Document made available under the Patent Cooperation Treaty (PCT)

International application number: PCT/US05/004176

International filing date: 11 February 2005 (11.02.2005)

Document type: Certified copy of priority document

Document details: Country/Office: US
                       Number: 60/544,501
                       Filing date: 13 February 2004 (13.02.2004)

Date of receipt at the International Bureau: 14 March 2005 (14.03.2005)

Remark: Priority document submitted or transmitted to the International Bureau in
           compliance with Rule 17.1(a) or (b)

1292629

THE UNITED STATES OF AMERICA

TO ALL TO WHOM THESE PRESENTS; SHALL COME:

UNITED STATES DEPARTMENT OF COMMERCE

United States Patent and Trademark Office

March 05, 2005

THIS IS TO CERTIFY THAT ANNEXED HERETO IS A TRUE COPY FROM THE RECORDS OF THE UNITED STATES PATENT AND TRADEMARK OFFICE OF THOSE PAPERS OF THE BELOW IDENTIFIED PATENT APPLICATION THAT MET THE REQUIREMENTS TO BE GRANTED A FILING DATE.

APPLICATION NUMBER: *60/544,501*
FILING DATE: *February 13, 2004*
RELATED PCT APPLICATION NUMBER: *PCT/US05/04176*

Certified by

[signature] Jon W. Dudas

Under Secretary of Commerce
for Intellectual Property
and Director of the United States
Patent and Trademark Office

Please type a plus sign inside this box

PROVISIONAL APPLICATION FOR PATENT COVER SHEET

This is a requ st for filing a PROVISIONAL APPLICATION FOR PATENT und r 37 CFR 1.53(c).

Express Mail Label No.	EL908622059US	*Elaine Akaka* Elaine Akaka

INVENTOR(S)		
Given Name (first and middle [if any])	Family Name or Surname	Residence (City and either State or Foreign Country)
Marc Victor	Gorenstein	313 Brookline Street Needham, MA 02492

[✓] Additional inventors are being named on the ___✓___ separately numbered sheets attached hereto

TITLE OF THE INVENTION (280 characters max)

TITLE

METHOD AND APPARATUS TO TRACK AND QUANTITATE CHEMICAL ENTITIES

Direct all correspondence to: **CORRESPONDENCE ADDRESS**

[ ] Customer Number

OR

Customer Number

Customer Number Bar Code

[✓] Firm or Individual Name	Anthony J. Janiuk, Esq.				
Address	Waters Corporation 34 Maple Street – MS: LG				
City	Milford	State	MA	Zip	01757
Country	USA	Telephone	(508) 482-2714	Fax	(508) 482-2320

ENCLOSED APPLICATION PARTS (check all that apply)

[✓] Part I — 16
Part II — 73

[✓] Figures
[✓] Abstract — 6 / 6

[ ] CD(s), Number

[✓] Other: Prepaid Return Postcard

[ ] Application Data Sheet. See 37 CFR 1.76

METHOD OF PAYMENT OF FILING FEES FOR THIS PROVISIONAL APPLICATION FOR PATENT

[ ] Applicant claims small entity status. See 37 CFR 1.27.

[ ] A check or money order is enclosed to cover the filing fees

[✓] The Commissioner is hereby authorized to charge filing fees or credit any overpayment to Deposit Account Number: **23-0503**

FILING FEE AMOUNT ($): **160.00**

The invention was made by an agency of the United States Government or under a contract with an agency of the United States Government.

[✓] No [ ] Yes, the name of the U.S. Government agency and the Government contract number are:

Respectfully submitted		
SIGNATURE		Date: **February 13, 2004**
TYPED OR PRINTED NAME	Anthony J. Janiuk, Esq.	REGISTRATION NO. (if appropriate): **29,809**
TELEPHONE	(508) 482-2714 Fax: (508) 482-2320	Docket Number: **04-388**

PROVISIONAL APPLICATION PATENT COVER SHEET

SECOND PAGE

ADDITIONAL INVENTORS:

Guo-Zhong Li, 14 Morse Street, Westborough, Massachusetts 01581

IDEA DISCLOSURE

Please provide the following information.

1.   Title of invention: Method and Apparatus to track and quantitate chemical entities

2.   Identify all person(s) who you presently believe were involved in development of the invention.

     Name: Marc Victor Gorenstein
     Address: 313 Brookline Street
                                   Needham, MA 02492
     Country of Citizenship: USA

     Name: Guo-Zhong Li
     Address: 14 Morse Street
                                   Westborough, MA 01581
     Country of Citizenship: China

     If there are additional person, please provide similar information for each.

3.   Identify all person(s) who you presently believe were involved in any reduction to practice of invention.

     Marc Victor Gorenstein
     Guo-Zhong Li

4.   What is the technical field of the invention?
     Chromatography. Mass spectrometry. On-line LC/MS separations.

Describe your present understanding of the invention.

Introduction
A key problem in analytical chemistry is the estimation of the concentration of one or more molecular entities contained within in a complex mixture.

Liquid chromatography (LC) followed by mass spectrometry (MS) is a well-known technique (LC/MS) that can separate large numbers of chemical entities in a sample, thereby facilitating the concentration measurement, or quantitation, of each. By measuring the exact mass of an entity, the MS can track the entity between samples. By measuring the response or intensity of the tracked entity, then the concentration of an entity can be tracked from sample to sample.

Here we consider, but do not limit, the discussion to samples that are separated by LC, ionized with electrospray ionization, and analyzed by mass spectrometers such as quadrupoles, time of flight, ion traps, or combinations of these analyzes. This discussion also pertains to entities that may be fragmented by MS-MS, or MS^n techniques.

In electrospray ionization, small molecules can produce a single characteristic ion. A larger molecule, such as a peptide or a protein, can produce a set of ions. Algorithms, well-known in the art, can reduce such sets of ions to a single effective ion. Thus we use the term entity to mean a single molecule whose concentration can be determined by examination by one ion or by more than one ions; in either case we assume that an effective mass and retention time can be assigned to each entity.

In LC/MS, a sample is injected into the system for analysis, so for each injection, the LC/MS system measures the retention time, molecular weight, and intensity of each entity.

Comparison of intensities of corresponding entities between injections is the basis of, for example, determining of the concentration of an entity changes significantly between control and unknown samples. Changes in expression level of a protein between samples will manifest itself by changes in the protein's concentration between samples.

A set of samples may be processed via sequential injections. The same sample may be injected multiple times to provide a set of replicate injections. As an example, one may inject each of two distinct samples (a standard and unknown) three times, to produce a total of six injections. From this data, the reproducibility of the concentration measurements can be inferred for each entity, as well as the change in concentration of each entity between the control sample and the unknown sample. Each sample may contain an amount of an internal standard to provide a relative calibration between sampled.

For a technique to determine the concentration of any entity, it must first adequately resolve that entity from all others. The LC/MS technique allows us to separate entities (or the ions associated with an entity) in both mass and retention time. Entities that co-elute in retention time, which would otherwise be indistinguishable) can be resolved in mass, thus allowing for an accurate estimate of their intensity.

The problem
But to associate, or to track, an entity from one injection to another, accurate mass alone may or may not be sufficient. It is this issue that this disclosure will address.

To see the issue, consider the properties of mass and retention time of a molecule.

The molecular weight is an intrinsic property of a molecule. A mass spectrometer measures the ratio of molecular weight to charge, m/z. We use the symbol μ to indicate the mass-to-charge ratio, thus $\mu \equiv m/z$. Values for μ can be compared directly between injections. Any variations in

measured values of μ between injections for the same entity must be due only to instrumental noise sources.

For samples such as peptides or proteins, electrospray ionization may allow us to determine the charge state, z, which then allows us to infer the molecular weight m of the entity. So we may track entities by their molecular weight. For the purposes of this disclosure we can use the empirically observed value for μ or the inferred value m, interchangeably.

With sufficiently high mass accuracy, each entity is potentially uniquely distinguishable based upon its value for μ. Thus with a sample containing few entities, a high accuracy mass spectrometer, such as a time-of-flight (TOF) analyzer with resolution of $m/\Delta m \approx 20,000$ and sufficient chromatographic resolution to simply separate the entities, each entity can be tracked from one injection to another based upon accurate measurements of μ alone.

[Note that we are using a molecules value for m or μ to track it between injections. We are not necessarily using m or μ to identify the entity in the sense of determining its chemical composition or structure. We are using the molecular weight as an empirical and possible unique, identifier of the chemical.]

It may be that mass alone is not sufficient to track an entity from one injection to another. If mass accuracy was low and the sample is complex, then it may be that the mass of an entity seen in one injection may match the empirically observed mass of an unrelated entity in another injection. For example, we may have two entities where μ is 1024.200 amu and 1024.300. These entities are distinguishable with MS resolutions less than 0.100 amu, but they will not be distinguishable with resolution greater than 0.100 amu.

The chromatographic retention time of an entity can be an additional, potentially independent identifier of that molecule entity. Now, a molecule's retention time is not an intrinsic property. Its value depends on the interactions of the molecule with the liquid and solid phases in the chromatographic separation, among other effects. But even though the retention time is not intrinsic, its value can be made highly reproducible for a given separation method. Ideally, if the retention time were exactly reproducible and to high accuracy, then the combination of agreement in both mass and retention time could well be sufficient to allow each entity to be uniquely tracked from one injection to another. That is, the likelihood that two different entities shared the exact same retention time and mass would be highly unlikely.

However, retention time is not exactly reproducible between injections. Retention time of a molecule can wander from injection to injection.

The prior art

But there can be regularities in retention time that we can take advantage of. An object of this disclosure is to show how certain regularities in retention time, combined with a high-mass accuracy MS can allow us to reduce or eliminate the ambiguity that may occur with comparisons of μ alone.

There are four regularities in retention time we consider. The first is well known in the prior art. Three are not known.

The first regularity is that if an entity elutes in injection A at time t, then that entity will elute in another injection, B within a window of width $t \pm \Delta t$. Though retention time is not exact, its wander is bounded from one injection to another. This bound can be determined empirically. We call this the coarse retention time window, Δt_c

Given such a bound, we can then ask: can two entities that elute within the bound have the same (i.e, measured to be the same) value for μ. If all entities that lie within Δt_c have different values

for μ , then we can track entities based upon similarities of retention time and matching value for μ .

In this disclosure, we consider samples that are complex enough that within the retention time window Δt_c there can still be significant number of entities whose values for μ do not render them unique.

Thus we consider a situation where most, but not all entities in a mixture have unique masses. The aim of this invention is to take advantage of the entities that do have unique masses, and to take advantage of these unrecognized regularity in chromatographic retention time, in order to uniquely track those remaining entities that might otherwise not be distinguished by mass alone.

Examples of such sample are digests of peptides that derive from natural protein samples. Peptide digests of blood serum, for example, can contain 10,000 or more distinct peptides. In a chromatographic separation, there can be 30 or more peptides elute within the width of a a chromatographic peak. The question then becomes whether we can count on the values of molecular weight associated with these 30 peptides can be relied upon to be unique in all cases.

The invention

The idea behind this disclosure is to take advantage of three additional, but previously unrecognized regularity of the retention time behavior of entities.

The second regularity occurs when two different chemical entities elute at the same identical retention time in all separation. That is, if they elute at the same time in one separation, then they elute at the same identical retention time in all other separations. The retention time of the pair may change from separation to separation, but the difference in retention time if zero in one separation will be zero between that pair for all separations.

This regularity does occur in the important case of peptide mixtures. Two peptides that elute at the same retention time in one separation will elute at the same identical retention time in all other separations. The retention time may change from separation to separation, but the difference in retention time if zero in one separation will be zero between that pair for all separations.

This disclosure then restricts itself to samples where we can count on this regularity: Two entities that elute at the same identical retention time in one separation will elute at the same identical retention time in all other separations.

The third regularity will, at first, seem at odds with the second. There is intrinsic measurement errors associated with retention time. Thus two entities that elute at the same retention time in all injections will in fact elute at somewhat different *measured* elution times. The *measured* retention times will match *only on average*. We can think of these as statistical errors associated with locating the top of peaks. Thus if an entity elutes at 10.0 min in an injection, its measured retention time might vary, by say up to +/-0.2 minutes. Normally, this intrinsic variation in retention time is masked by the retention time wander between injections. This intrinsic variation is measured when we track two entities that both elute at 10.0 minutes in an injection.

Generally this statistical error is much less than the wander error, described by Δt_c . We call the threshold associated with the statistical measurement error, the fine retention time threshold Δt_f

A fourth and final regularity occurs for entities that elute closely in time, but not exactly the same retention time. The retention-time order at which those two entities elute may change from separation to separation. However, if there is a third entity that elutes between those two it will always elute between those two.

For example, as a result of retention time wander the time offset between two entities may change from injection to injection. For example if the entities elute at 2.0 and 2.4 minutes in one injection, they may elute at 2.5 and 2.7 minutes in a second injection. While it is true that the first entities retention time drifted by 0.5 minutes between the injections, it is not this quantity that we are interested in. Rather, it is the difference in retention times between entity one and two. That difference was 0.1 minutes in the first injection and 0.2 minutes in the second injection. The third regularity pertains to a third entity that elutes in injection 1 between these two times. Let's say that it elutes at 2.1 minutes. The third regularity is that if the third entity elutes between 1 and 2 in injection one, then it will also elute between 1 and 2 in other injections. Moreover, the offset is proportional. Thus in injection 2 the entity will elute at 2.55 minutes.

Regularity 1 and 3 must exist in all chromatographic separations; they are the characteristics of a reproducible measurement, or a robust method. Regularities 2 and 4 may or may not occur for all entities in a complex a mixture. But they do occur for peptide digests, and likely hold for mixtures where the entities have related chemical interactions with the chromatographic stationary and moving phases. The method described here can recognize when these regularities occur, and when they do, this method can take advantage of them for the purpose of tracking entities from injection to injection.

The method to be disclosed takes advantage of these regularities to assign each entity a reference retention time. This reference retention time is unique in the sense that if two entities do not have the same reference retention time, they cannot be the same entity. If they do have the same reference retention time, they can be the same entity.

Entities are then tracked by requiring that they have the same molecular weight and the same reference retention time. Entities that differ significantly in either or both molecular weight or retention time are not the same.

To summarize: In complex separations, more than one entity may have the same molecular weight, to within the ability of the instrument to distinguish. This invention describes makes use of accurate mass measurement and hitherto unrecognized regularities in retention time to determine a retention time map. The map then allows the assignment of a reference retention time to each entity in a separation. The reference retention times of entities can then be compared between separations.

The Method

The method disclosed here assigns a reference retention time to each entity in each injection. It is the reference retention time that can be directly compared and thus can be used to track entities between injections

Consider two injections A and B. The method compares entities in A to those in B. From the results obtained from this comparison, the method assigns reference retention times to entities in injection B. Given a third injection, C, the method compares entities in A to those in C to obtain reference retention times for C. The reference retention times assigned to entities in B and C can then be directly compared to each other and/or to the retention times in A. The method, in effect, removes the effect of retention time drift between injection B and A, and between C and A for each entity in B and C.

The method can then be extended to as many injections of as many samples as desired.

In summary, the method first uses a subset of entities in A and B to obtain a retention-time map. It is from this map that a reference retention-time is obtained for all entities in B.

Given injection A and B, the method for determining the retention-time map between A and B as follows:

1) We use a subset of entities in injection A and B to first construct a retention time map. It is this map that will be used to obtain the reference retention times.

2) We can choose subsets of entities in A and B based upon their intensity. For example, we could consider entities above a threshold-intensity. In the preferred method, a threshold is applied to entities in A, and that threshold is the median intensity of intensities in injection A. Similar, we consider for injection B all entities whose intensities lie above the median intensity in injection B.

3) We could normalize the intensities from injection A or B either before or after applying a threshold-intensity.

4) To construct the retention time map, we choose a coarse retention time threshold Δt_c. The preferred value is +/-5 minutes. The value can be refined upon examination of the mapping results. It is this value that is describes the maximum wander that occur in retention time. This value will be confirmed and possible refined in a later step.

5) Choose a molecular weight threshold Δm. The threshold could also be expressed as parts per million $(\Delta m/m) \times 10^6$. We could also choose a m/z threshold $\Delta \mu$. The method disclosed here assumes that this threshold has been obtained through knowledge of the properties of the MS. This threshold can be obtained by methods known in the art.

6) We then perform a search that compares all threshold-selected entities in injection A to those in B. This search finds those entities in A that have a single match to a threshold-selected entity in injection B. Two entities match if the difference in their mass falls below the mass threshold Δm AND if the difference in their retention time falls within the coarse retention time threshold Δt_c AND if there is only one entity in B that meets that criteria AND if the intensity of both entity lie above the respective median intensities. Such search methods are well-known in the art.

7) This map will then contain only pairs of entities that have unique matches in molecular weight and in coarse retention time and match any intensity requirements

Thus we have obtained a set of N pairs of entities, each indicated by a subscript i. Each pair satisfies the properties:

$$\left| m_i^B - m_i^A \right| < \Delta m$$

$$\left| t_i^B - t_i^A \right| < \Delta t_c$$

$$I_i^A < median\left(I^A \right)$$

$$I_i^B < median\left(I^B \right)$$

It is within the scope of this method to add other restrictions. For example one could require that the intensity ratios satisfy fall within a threshold. Thus the result of search would produce entities that satisfy.

$$\frac{I_i^A}{I_i^B} < r \text{ and } \frac{I_i^A}{I_i^B} > \frac{1}{r}$$

a preferred value for r might be 2. This additional intensity restriction is optional and is not required by the method.

If one is comparing ions of known charge state, one could require that the charge states match. Thus the result of the search would produce entities such that $Z_i^A = Z_i^B$

[Figure 1a,1b] Figure 1a,b plots the results of this coarse search. The quantity $\Delta t_i \equiv t_i^B - t_i^A$ is plotted on the y-axis versus t_i^B on the x-axis. Note that most of the points cluster along a dense backbone. There is scatter about the backbone and there are outliers.

Given this list of matched pairs, the next step is to construct the retention time map. To do this, we sort the list of so that the values for t_i^B are in ascending time order, thus $t_{i+1}^B > t_i^B$ for $i = 1, \ldots, N-1$. These pairs result from the coarse search.

Examination of this plot in Figure 1 confirms the validity of the choice of the value for Δt_c. It also can suggest a refined value; e.g., for small amplitude excursion, the value for Δt_c could be reduce. If it appears that the excursion exceed the initial value for Δt_c, the value could be included, and steps 1-6 can be repeated for a larger value for Δt_c.

8) The next step in obtaining the map is to filter the backbone. That is, we wish to find a refined value for $\Delta t_i \equiv t_i^B - t_i^A$, as a function of t_i^B. We could do this by applying a moving average filter, replacing each value for Δt_i with a weighted average of its neighbors. However, since there are outliers, we employ a median average filter. Thus we replace each value of Δt_i with the median value of itself and its M nearest neighbors. Typical value for M will be at least 5 points, or as many as 20 for the data sets we consider. **[Figure 2.]** Figure 2 shows the filtered results for a 5 point median filter. We see how the outliers are removed by the median filter.

We now have a set of values Δt_i^m and t_i^B, which are the median filtered values. We now obtain

$$t_i^{B_{ref}} \equiv \Delta t_i^M + t_i^B$$

We now have N pairs of values, $t_i^B, t_i^{B_{ref}}$. It is these pairs that are the retention time map. The retention time map is then described as a point to point look-up table, which is described by these paired values. The retention time map is obtained from a subset of entities. The map is central to the determination of the reference retention time for all entities in B.

Given this map, we assign reference retention times as follows.

The entities in B then fall into two categories, those that are part of the look-up table, and those that are not.

9) For the entities in B that are part of the look up table, the reference retention time is simply $t_i^{B_{ref}}$ as given above.

10) For the entities that are not part of the look up table, we make the assignment based upon a linear interpolation:

$$t_k^{B_{ref}} = t_i^{B_{ref}} + \left(t_k^B - t_i^B\right)\frac{t_{i+1}^{B_{ref}} - t_i^{B_{ref}}}{t_{i+1}^B - t_i^B}$$

where $t_{i+1}^B > t_k^B \geq t_i^B$, and were the subscript i and $i+1$ specifies entries in the retention time map, i.e.,g in the look-up table. The entities specified by the subscript k are entities not in the look-up table. It is this equation that specifies how reference retention times are found, and is the central object of this invention.

11) For each entity in A, it is convenient to define its reference retention time is simply its retention time. Thus $t_i^{A_{ref}} \equiv t_i^A$, for all entities in A.

We have now assigned reference retention times for each entity in A and in B. The above procedure has removed the retention time offset between entities.

12) Given an injection C, we repeat the steps above, replacing B with C to obtain $t_i^{C_{ref}}$

We can now track all entities between A and B by applying the following method.

13) We choose a fine retention time threshold, Δt_f This fine reference retention time can be obtained from histogram techniques that make use of the reference retention times of entities in B that are not part of the map. Typically, the fine value for retention time can be 0.4 minutes. Thus, we have reduced the retention time threshold from 5 to 0.4 minutes by this method. This in turn has the effect of reducing or eliminating ambiguities in comparing entities having the same molecular weight.

[**Figure 3**] Figure 3 shows the values that satisfy the fine retention time criteria, $\left|t_i^{A_{ref}} - t_i^{B_{ref}}\right| < \Delta t_f$, after the look-up table (the backbone) is established.

We can now attempt to track all entities in A, B, C, and all other injections. The fine offset Δt_f is used for the final search.

14) For example, we can compare all entities in B to all entities in A, and retain only those that meet the following tracking criteria:

$$\left|m_i^A - m_j^B\right| < \Delta m$$

$$\left|t_i^{A_{ref}} - t_j^{B_{ref}}\right| < \Delta t_f$$

The search is over any entity (indexed by i) in injection A versus any entity (indexed by j) in injection B. Note that the mass window criterion is unchanged. But the retention time criterion is changed. We compare the *reference* retention times to the fine search windows. We have a match if both criteria are met. No intensity criteria need be applied, though one could be optionally applied.

As mentioned, an additional injection can be accommodated by comparing all entities in C to all entities in A, and retain only those that meet the following criteria:

$$\left|m_i^A - m_j^C\right| < \Delta m$$

$$\left|t_i^{A_{ref}} - t_j^{C_{ref}}\right| < \Delta t_f$$

or by comparing all entities in C to all entities in B, and retain only those that meet the following criteria:

$$\left| m_i^C - m_j^B \right| < \Delta m$$

$$\left| t_i^{C_{ref}} - t_j^{B_{ref}} \right| < \Delta t_f$$

Note that even though A is used as the common target for the retention time reference computation, reference retention times can be then compared between any two injections, such as between C and B. Thus a completely symmetric comparison for the purpose of entity tracking is provided by this method.

Entities can then be tracked as follows: two entities are the same if the have the same molecular weight (within a prior specified error) and if they have the same reference retention time (to within a prior specified error). The errors can be determined from within the properties of the data.

Traditionally, tracking is done by including internal standards to align retention times. These internal standards are chemical entities of known mass and injected at known concentration. The method proposed here does not require such internal standards. The method disclosed here does not have to know, a priori, which entities appear with unique masses. In effect the exact mass measurements allow us to use the subset of those entities that define the map to each act as a local retention time standard.

The invention is a method to define and specify reference retention times for entities. The uses of this invention are entity tracking between injections. The use of entity tracking then allows the analyst to quantify or track relative changes in concentration of entities between samples in a sample set.

The assignment of reference retention time requires that there be a coarse and a fine retention time threshold. The coarse threshold is the not to exceed. The fine threshold is the variation about zero. All unique mass hits at high SNR are found within the coarse threshold.

Applications of quantitation

Once an entity is tracked from injection to injection, the quantitative changes in concentration of the entity between samples can be measured. The quantitative response is the response as measured by the LC/MS system for the ion or set of ions that define an entity.

For example consider an experiment that consists of N replicate injections for each of M samples. The mean, median, standard deviation coefficient of variation can be obtained for mass, intensity and retention time for all entities tracked within each subset of N replicate injections. The mean of these quantities can be similarly tracked for each entity between the M samples.

The response of each entity as a function of sample can be input to standard statistical analysis such as SIMCA (Umetrics, Sweeden), or Pirouette (Infometrix, Woodenville, Washington, USA) can take as input the list of tracked entities produced by this method and reveal changes in entities between sample populations. The SIMCA and Pirouette packages and other software systems provide principle component analysis, or factor analysis techniques that can be applied to these data.

In particular, tryptic peptides that are digestion fragments of a common protein will have their intensities change in concert from sample to sample. Consider the following: one sample or set of samples contains a protein that is expressed at one level, and another sample or set of samples contain the same protein but now is expressed at a different concentration level. If tryptic digestion is performed, then the concentration of the tryptic peptides associated with that protein will scale from one sample to another. That is, the concentration patter will form one distinct

pattern in one sample, and will from a similar pattern in another sample, but with intensities values scaled overall to be larger or smaller, in response to a larger or smaller concentration of the parent protein.

Such correlated changes in concentration can be readily seen by factor analysis methods or by methods based on principle component analysis. Such a method can be used to identify the parent proteins whose concentration, or expression level, has changed from sample to sample. That is if a set of peptides produce a distinctive signature in a PCA plot, if those peptides point by to a common parent protein, then the protein whose expression level has change has been identified.

A definitive identification can be made by taking the exact mass of these associated peptides (the ones that change in concert) and identifying them using standard peptide fingerprinting software, such a provided by Peptide mass fingerprint software offered by www.matrixsciences.com or prospector.ucsf.edu

5.   What problem(s), if any, now known to you does the invention solve or attempt to solve?
The problem the invention solves is the tracking of chemical entities in a complex mixture from injection to injection. Given this ability to track entities, one can then track and trend changes in the responses, or intensities, and, ultimately concentrations of each entity. The ability to track concentration is key to discovering changes that may occur between control and unknown samples.

6.   Describe any now envisioned commercial applications for the invention:
This method may be employed in the Ion mapping software product (under development) and in the MarkerLynx product that is now commercialized. Additional products may be Waters Empower software or future software products.

7.   Conception and Disclosure

    a.   Date you presently believe the invention was first conceived: June 20th, 2003.
    b.   Date and form of what you presently believe is the first written description of invention: October 23rd 2003
    c.   Date and circumstances of what you presently believe is the first oral disclosure of the invention to another: July 2003
    d.   Identify all publications or other documents that you presently believe disclosure or describe the invention: None
    e.   Has a model or prototype been constructed?   Yes__X__   No____
    e.   Dates model/prototype were commenced and completed: July 15, 2003

8.   Testing and Reduction to Practice

    a.   Date of first test: July 15, 2003
    b.   Witness(es), if any: Scott Geromanos, Tim Riley, Jeff Silva
    c.   Date and description of what you presently believe was the first reduction to practice of the invention: July 15, 2003

Page 1 of 2   Signatures: _____

               Supervisor: _____

9.   First Sale of Offer For Sale of a Product or Process Embodying the Invention
Not applicable.

   a.   Date and circumstances of what you presently believe was the first offer for sale:
   b.   Date and circumstances of what you presently believe was the first sale:
   c.   Date and circumstances of what you presently believe was the first public use and/or demonstration of invention:
   d.   If the invention has not yet been used, date and circumstances of any presently planned public use or demonstration:

10.   Identify any prior publications or patents or other work now known to you that, in your present opinion, relate to the subject matter of the invention.

11.   A Third Party Rights

   a.   If any work relating to the invention was performed under a contract of funding arrangement with any governmental agency, please attach a copy of any documents relating to that contract of funding arrangement.
   b.   If any work relating to the invention was performed under an employment or consulting agreement, please attach a copy of such agreement.

12.   Attach copies of the following materials, where applicable, relating to the invention and its salient Features: photographs, engineering notebook excerpts, blueprints, videotapes, test results.

_____   _____
Inventor Signature   Date

_____   _____
Inventor Signature   Date

Witnessed by

_____   _____
Witness Signature   Date

_____
Witness Name

_____
Witness Address

Page 2 of 2   Signatures: _____

                Supervisor: _____
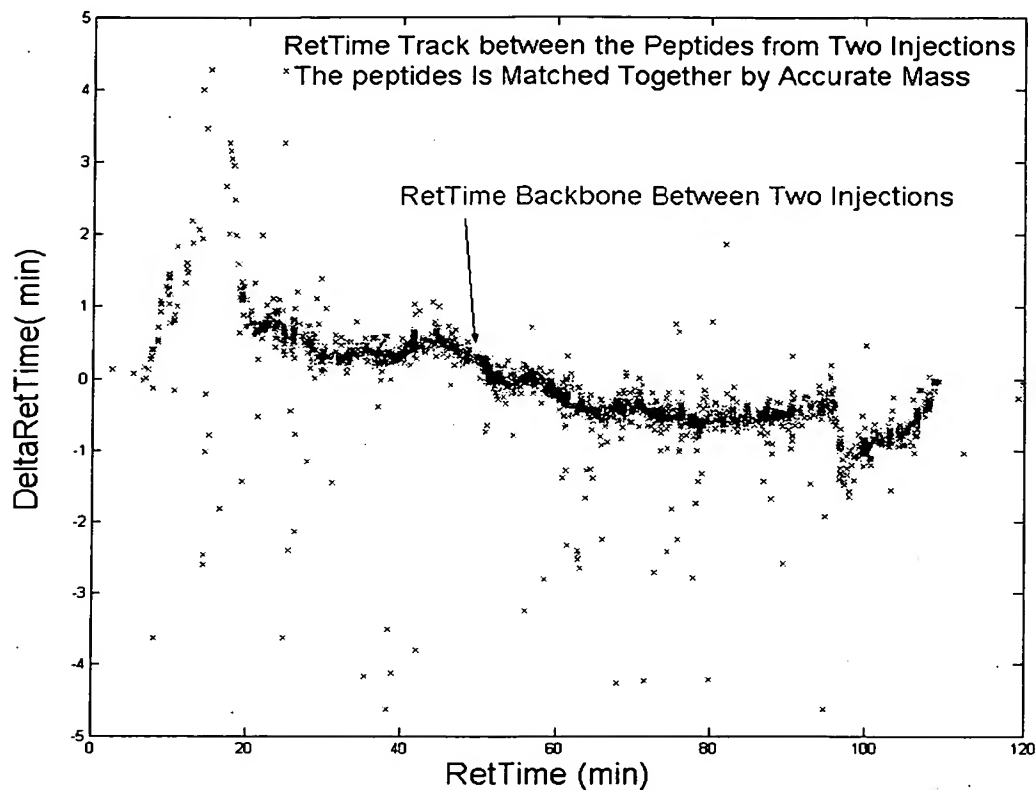
FIGURE 1a. The results of the coarse search. The horizontal axis is retention time. The vertical axis is difference in retention time. Points are plotted only if entities have masses the agree within in mass window of 0.020 amu **and** have retention time differences that are within +/-5 minutes. This +/-5 mintues is the coarse retention time search.
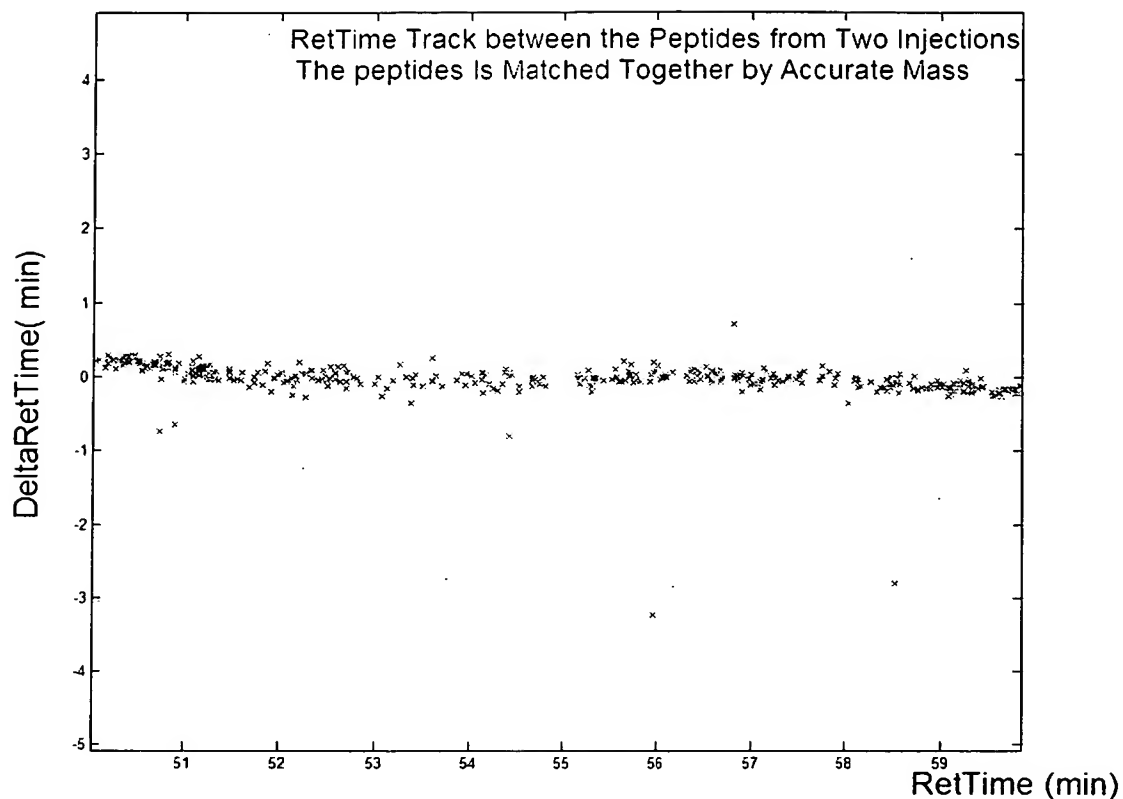
FIGURE 1b. The results of the coarse search. The horizontal axis is expanded retention time axis to how concentrated the matched pairs are on the vertical axis. This concentration reveals the backbone which is the basis of the retention time map.. The vertical axis is difference in retention time. Points are plotted only if entities have masses the agree within in mass window of 0.020 amu **and** have retention time differences that are within +/-5 minutes. This +/-5 mintues is the coarse retention time search.
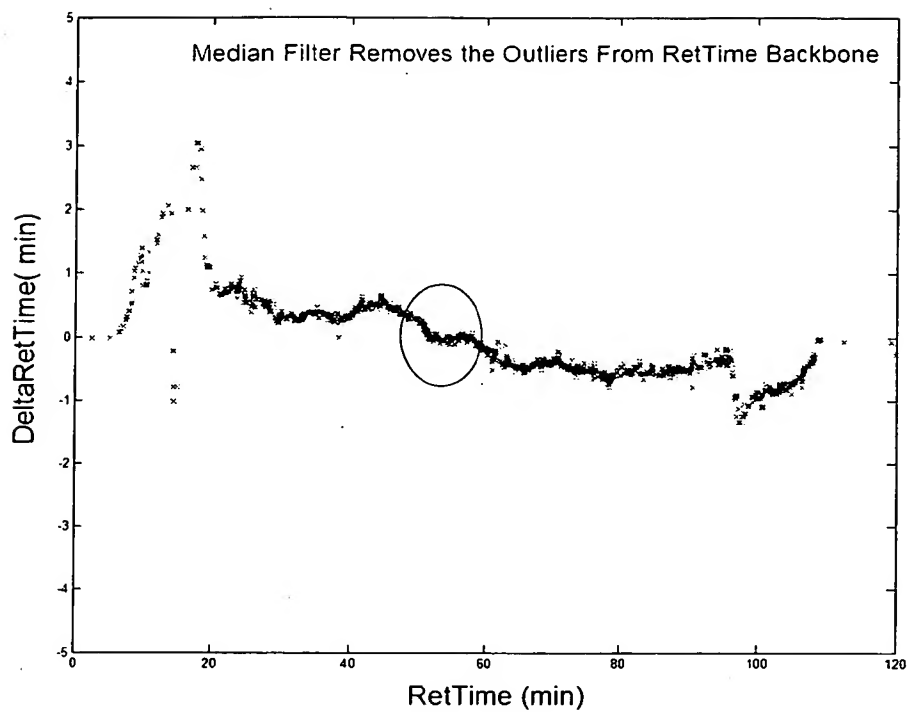
Figure 2a. After backbone selection by median filter and application of the fine retention time criteria.
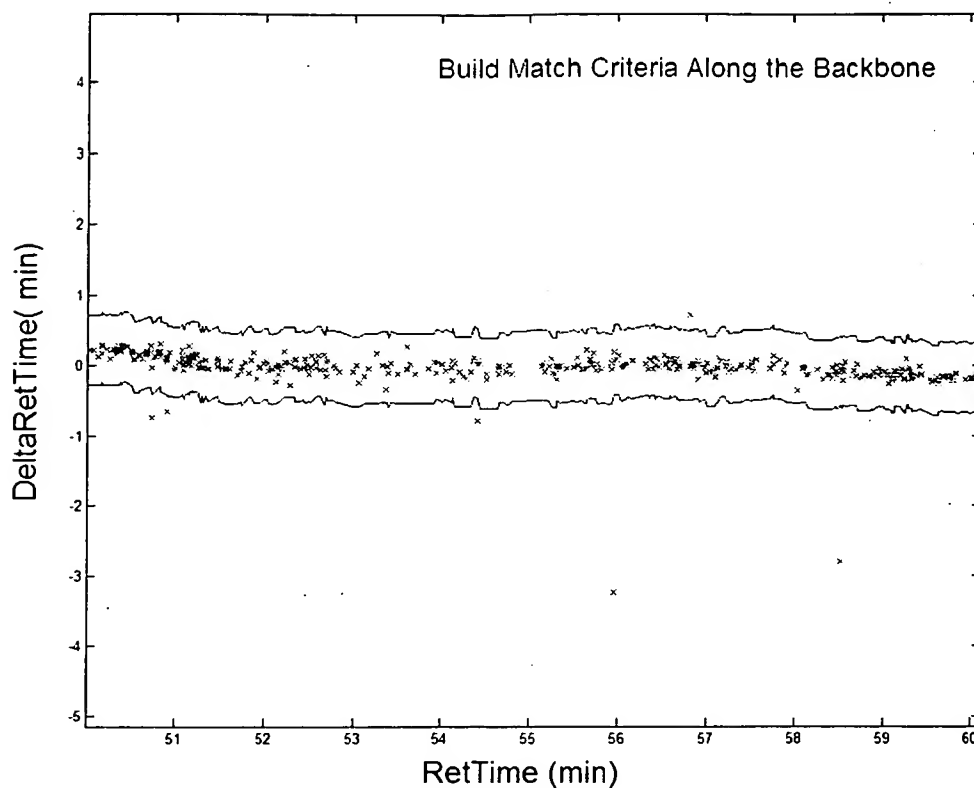
Figure 3, example of application of fine retention time. The solid blue line shows the fine retention time plotted versus retention time. Notice the fine retention time is +/- 0.4 minutes in this example. +/- 0.4 minutes is smaller than +/-5 minutes used in the coarse search. The backbone line is not drawn, but it is easily inferred from the limits of the fine search. The backbone is the blue line that would lie midway between the upper and lower limits of the fine search, drawn above.

Track3D (Track LC/MS Injections)

Guo-Zhong Li, Marc V. Gorenstein

(Software Algorithms Group)

- Purpose

- One Injection

- Two injections

- One Sample: Three Injections

- Two Samples: Six Injections

- Summary

Purpose

- Novel way to track LC/MS peaks in retention time

- Application to high accurate LC/MS

- Examples of ion mapping

LC Conditions

- *A Waters CapLC HPLC System* at 5.0 microliters/min flow rate.
- The column: **350 micron X 150cm AtlantisTM Column with 5 micron particles.**

- The gradient: from 1 – 30% B in 100 minutes (0.3%/min), 30-90% in 0.1 minute, and then held isocratic at 90% B for 10 minutes before returning to initial conditions for 20 minutes prior to the next injection.

- Solvent System A = 0.1% Formic Acid, 5% Acetonitrile
- Solvent System B = 0.1% Formic Acid, 95% Acetonitrile

MS Conditions

- *A QTof Ultima mass spectrometer in Vmode* (~12K FWHM)
- *a nano-LockSpray ion source* (~5ppm mass accuracy).
- The data was acquired *(LE/HE Switch)* in alternate scanning mode with a data acquisition time of 1.85 second for both the low (CE = 8) and high energy (CE = 28-35) channels.
- The inter-scan delay was set to 0.15 seconds. Data was collected over the *m/z* range 50-1990.

3

Sample Preparation

- Gilar Protein: bovine serum albumin,
  alcohol dehydrogenase (S. cerevisiea),
  Enolase (S. cerevisiea),
  Bovine Hemoglobin (a-Hb),
  Glycogen phosphorylase B
  (Rabit, PHS2)

Tryptically digested and prepared in equimolar mixtures.

Human serum tryptic digest

+

Tryptically digested gilar protein (0.1 pmoles)

LC/MS Raw Data is from Ion Mapping Project
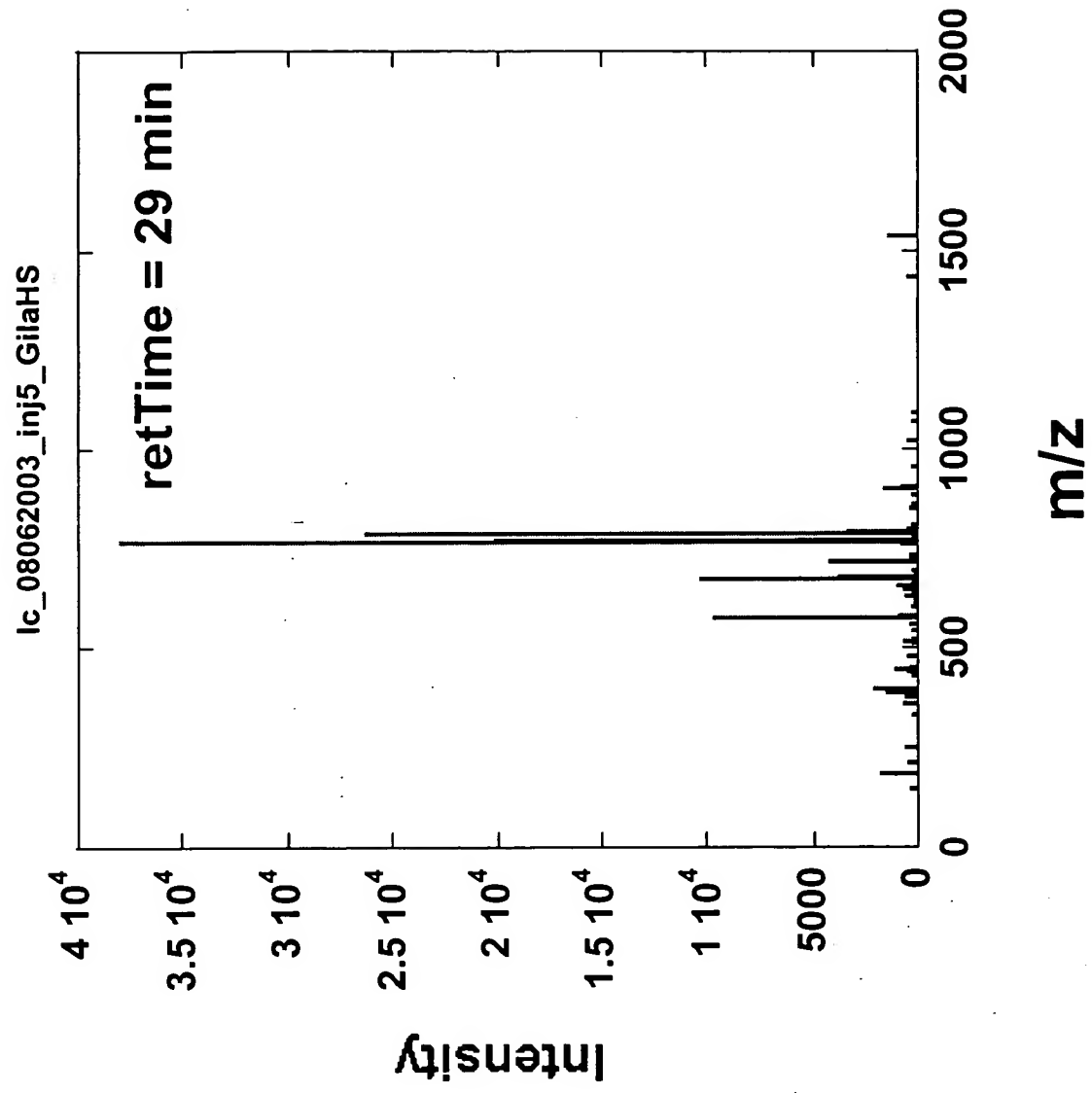
What is Ion Mapping?

Experiment
(ms, ms/ms)
mhp, retT, Intensity

Ion Mapping

Protein Database
(1 M Proteins)
Pep_mhp,
b & y ions,
hydrophobicity
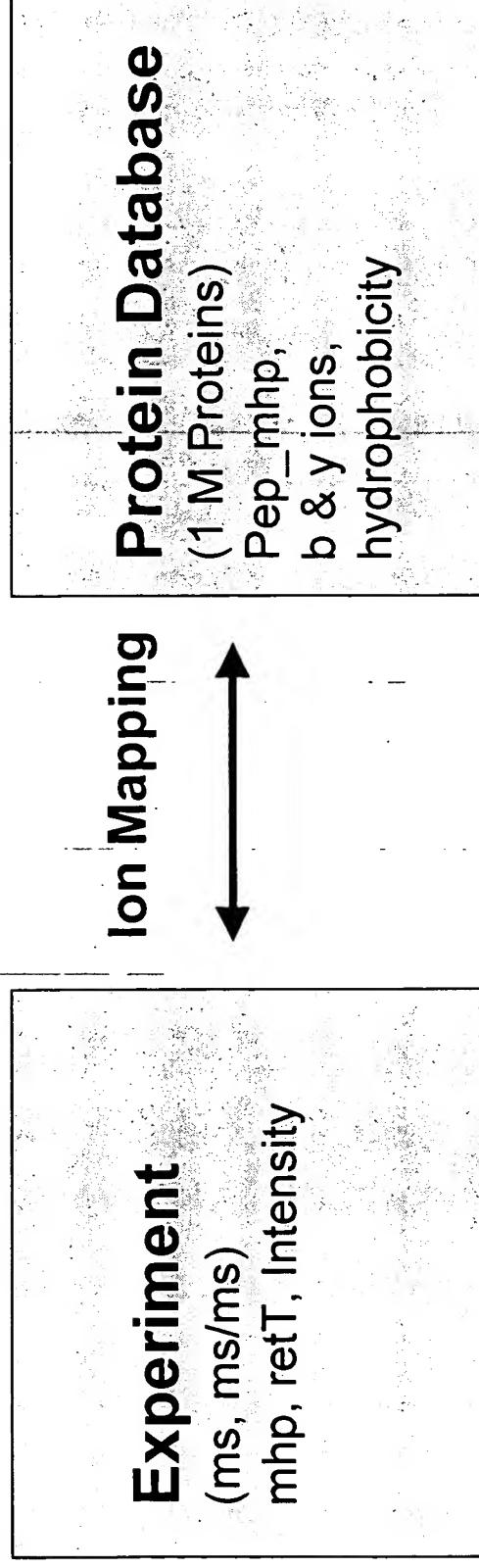
Key technology: Q/TOF, LE/HE Switch and RetTime

One Injection

Ic_08062003_inj5_GilaHS (Ion Mapping Project)

7

Ic_08062003_inj5_GilaHS

retTime = 29 min

Intensity

m/z

8
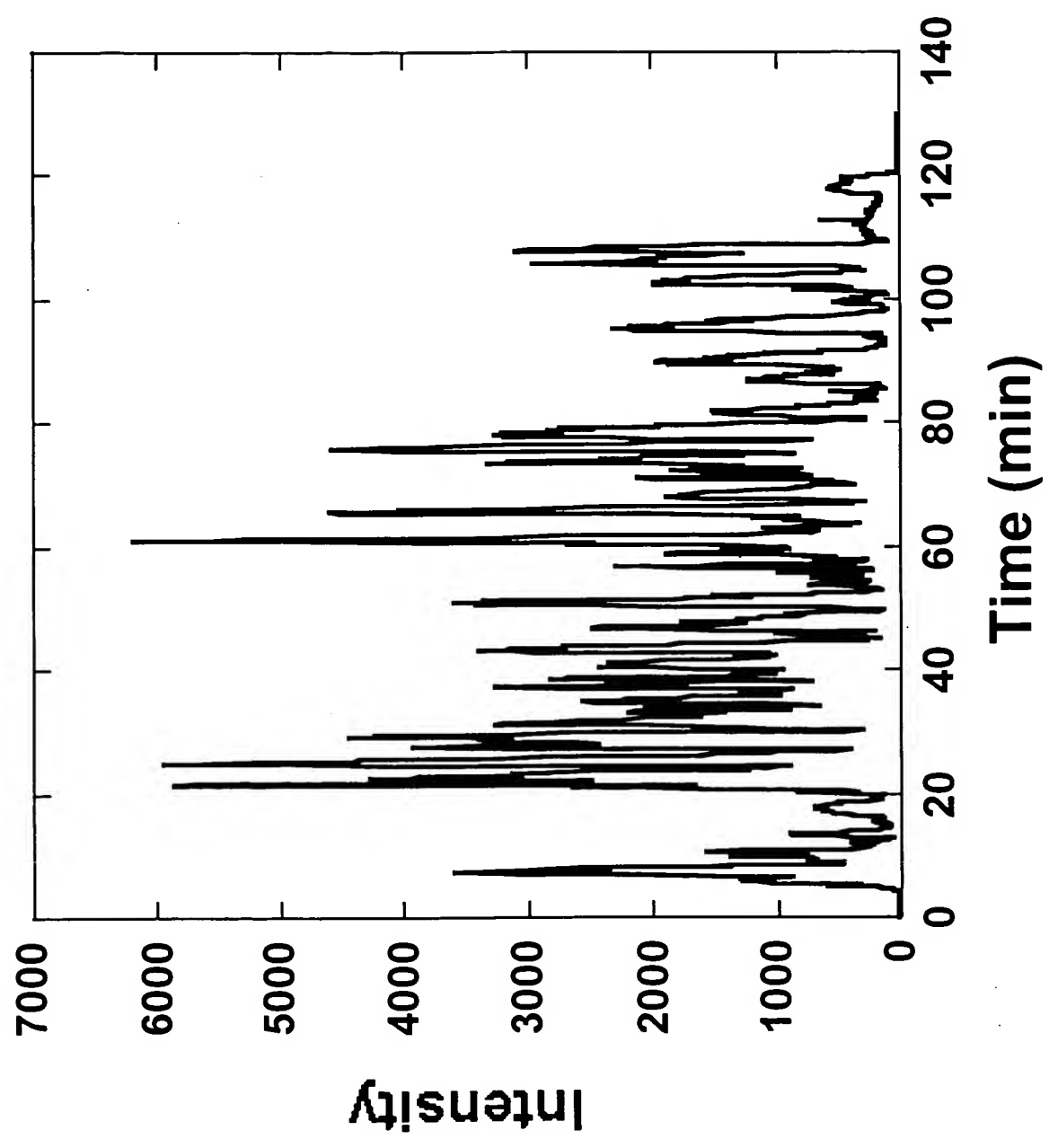
Data Processing

- Apex and The Unique Mass List Generator (UMLG) was used for the detection of accurate mass retention time (AMRT) components in all the studies.

- Track3d was used to match accurate mass-retention time components detected in different data sets and

- PIBHEC was used as a data base search algorithm for the qualitative identification of the detected peptides. All three software packages were releases received in July and August, 2003.

- PIBHEC search criteria used in the identification studies was as follows:
  The measured mass accuracy of low energy ions was required to be within +/- 10ppm of theoretical.
  The measured mass accuracy of high energy ions was required to be within +/- 25ppm of theoretical.

- Zero or one missed cleavage was allowed.

- A minimum of four contiguous y" ions excluding the y" one ion was required to co-elute in the high energy data with the low energy parent ion.

LC/MS Raw Data → Ion Sticks (m/z, retTime, Intensity)

Ion Mapping Application

• Reduced ions (*m/z, retTime, Intensity*)

to peptides (*mhp, retTime, Intensity, charge*)

by charge and Isotopic deconvolution.

Each peptide

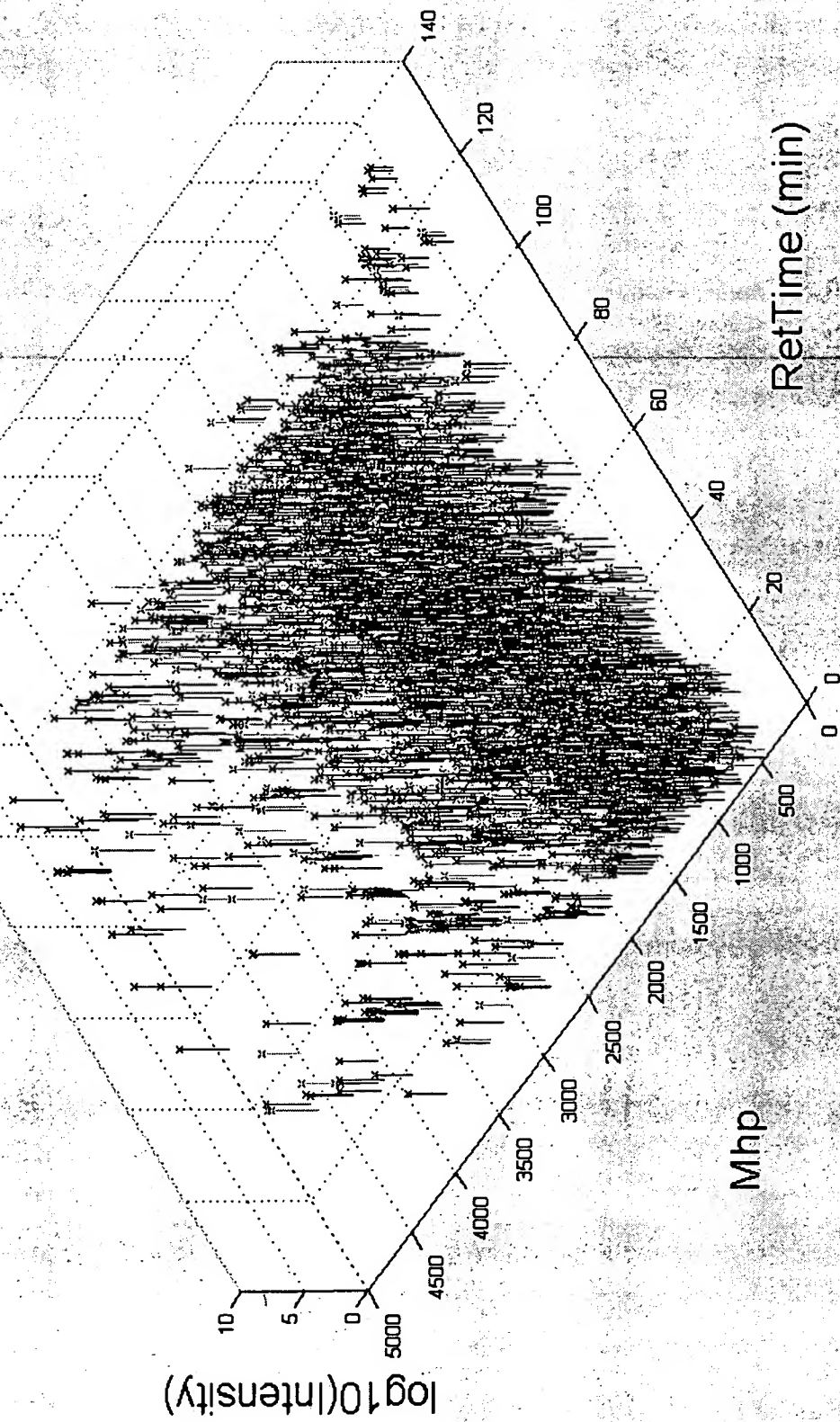– mhp reproducible to 10 ppm

– retTime can wander +/- 5 min

Peptide Sticks

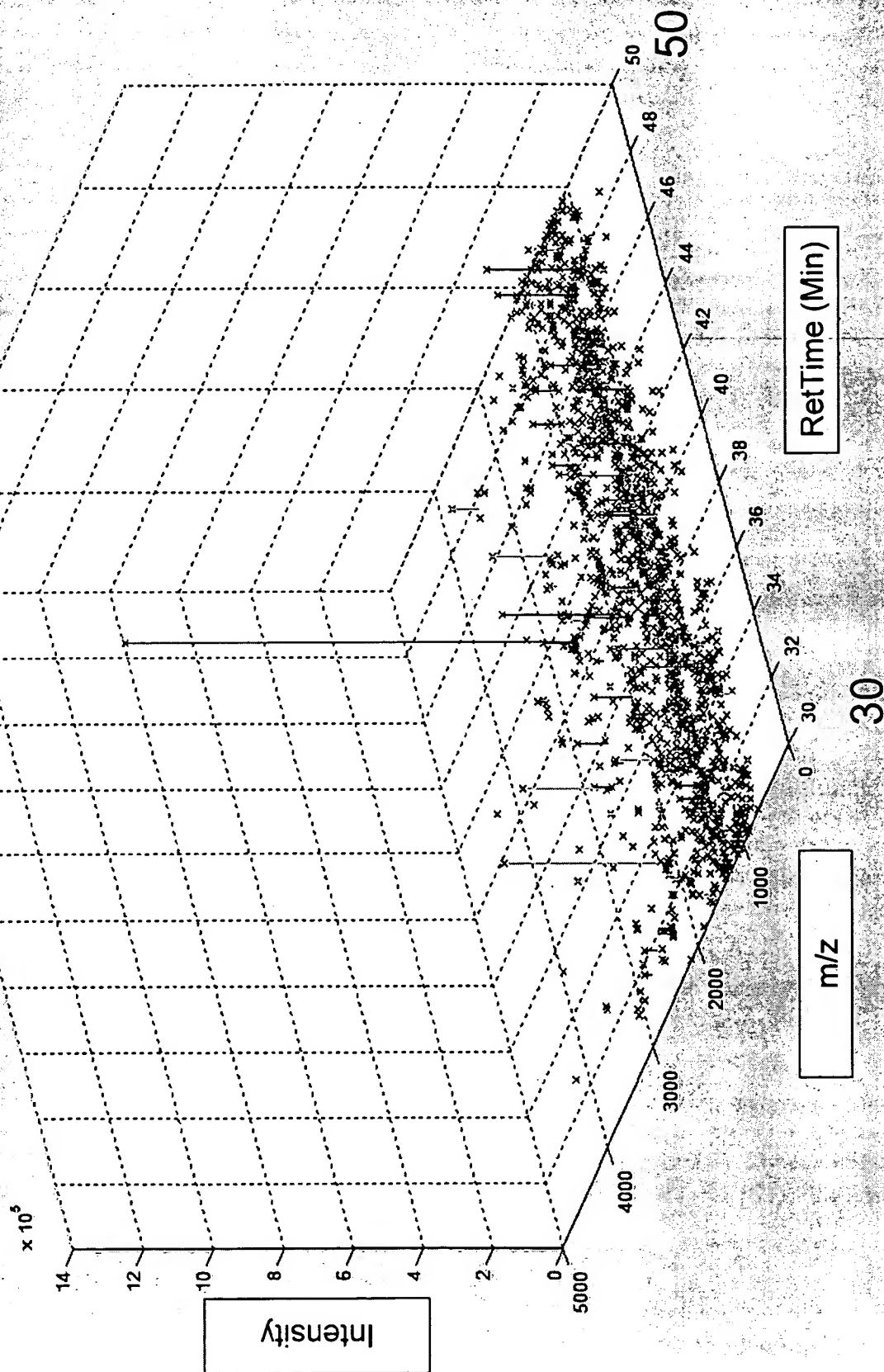## 3D LC/MS Peptide Data (Mhp, RetTime, Intensity) of Human Serum and 0.1 pmoles Gilar Proteins

RetTime (min)

Mhp

log10(Intensity)
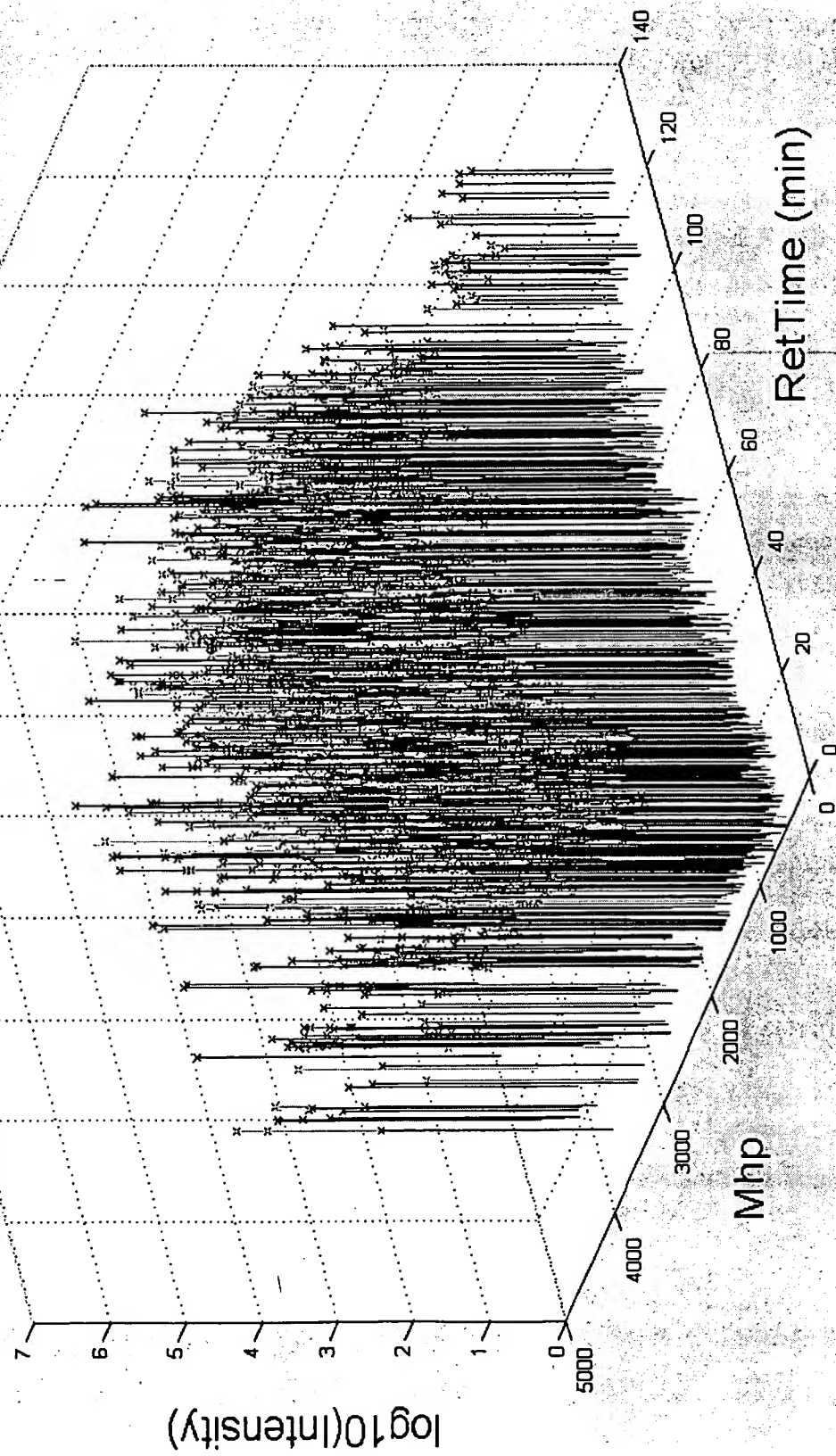
Peptide Sticks (Another Angle)

3D LC/MS Peptide Data (Mhp, RetTime, Intensity) of
Human Serum and 0.1 pmoles Gilar Proteins

RetTime (min)

Mhp

log10(Intensity)

LC/MS LE Sticks

Intensity

x 10⁵

14
12
10
8
6
4
2
0

5000
4000
3000
2000
1000
0

m/z

30   32   34   36   38   40   42   44   46   48   50

RetTime (Min)

30   50

Next slide show 3D Histogram plot:

retTime, mhp, number of peptide frequency distribution

Key point:

there is only one peptide in a (mhp x retTime) bin

2 Dalton

Number of Peptides in a (Mhp X RetTime) Bin
1 Bin = 2 Dalton X 0.5 minute
Human Serum and 0.1 pmoles Gilar Proteins (5 proteins)
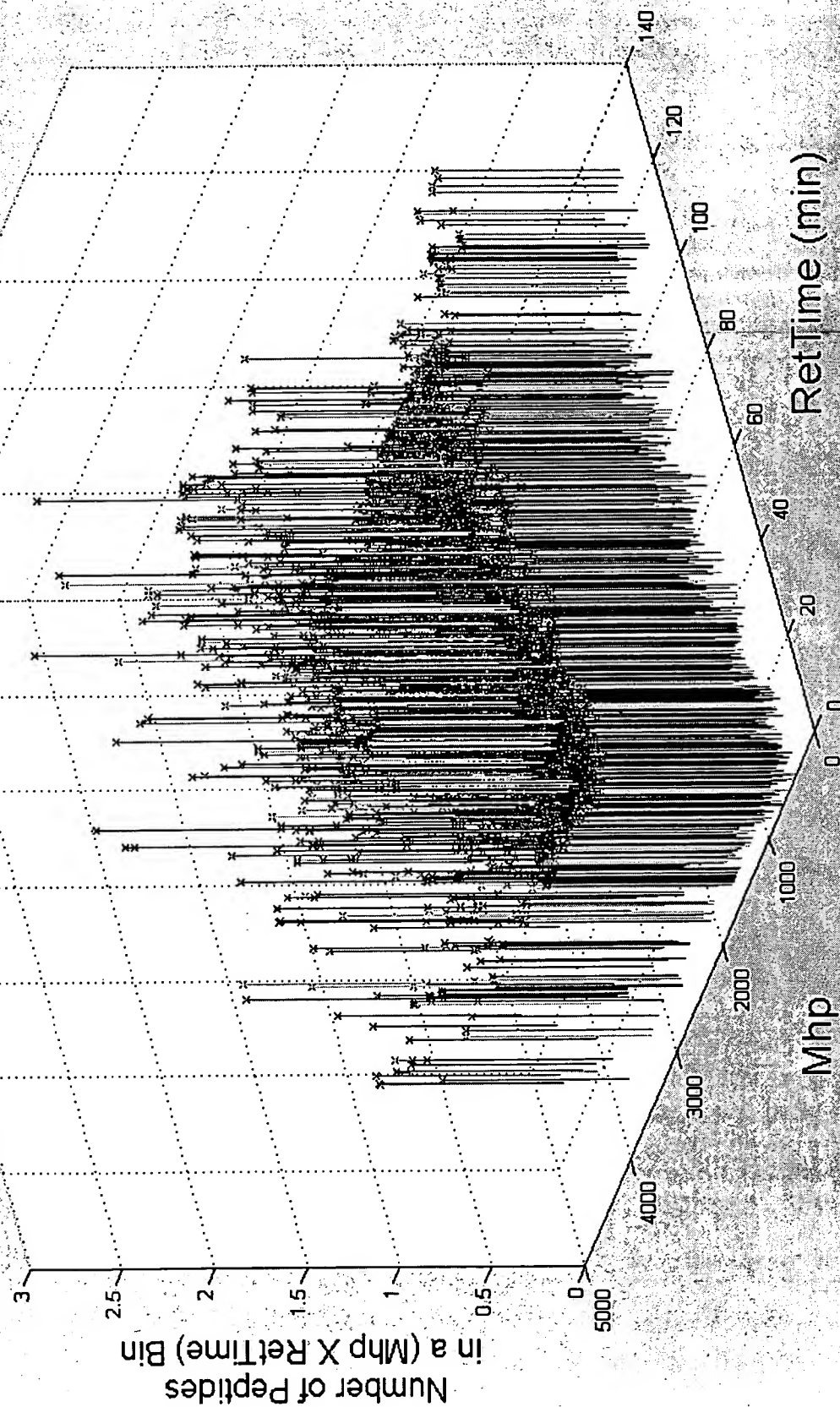
0.1 Dalton

Number of Peptides in a (Mhp X RetTime) Bin
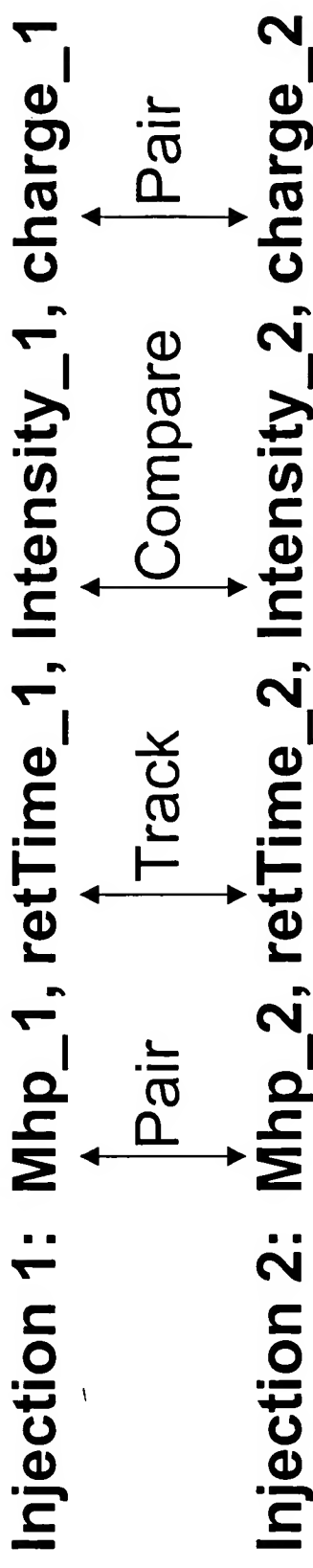### 1 Bin = 0.1 Dalton X 0.5 minute
Human Serum and 0.1 pmoles Gilar Proteins (5 proteins)

Axes: Mhp; RetTime (min); Number of Peptides in a (Mhp X RetTime) Bin
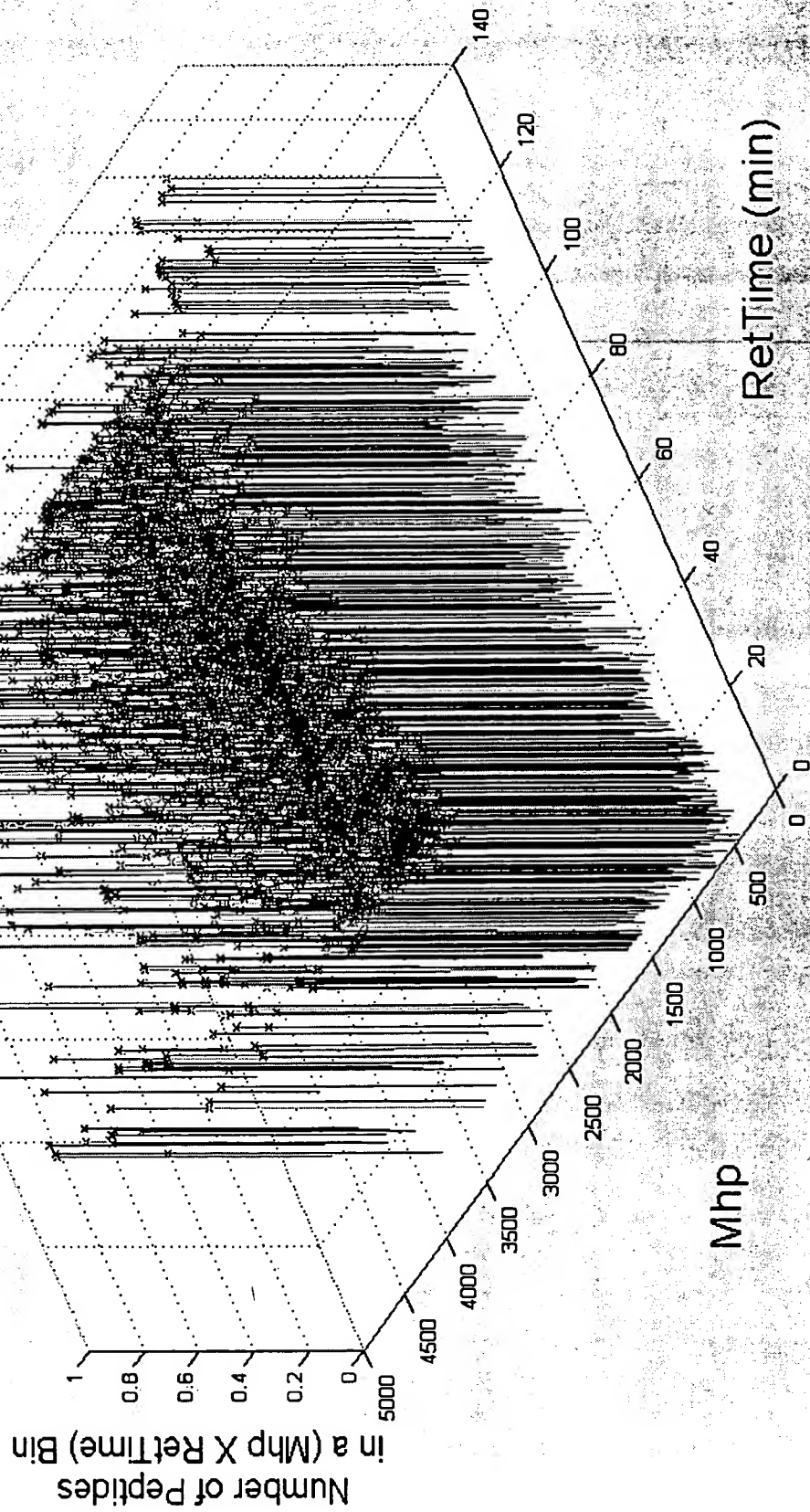
0.02 Dalton

Number of Peptides in a (Mhp X RetTime) Bin
1 Bin = 0.02 Dalton X 0.5 minute

Number of Peptides in a (Mhp X RetTime) Bin

m/z

RetTime (min)
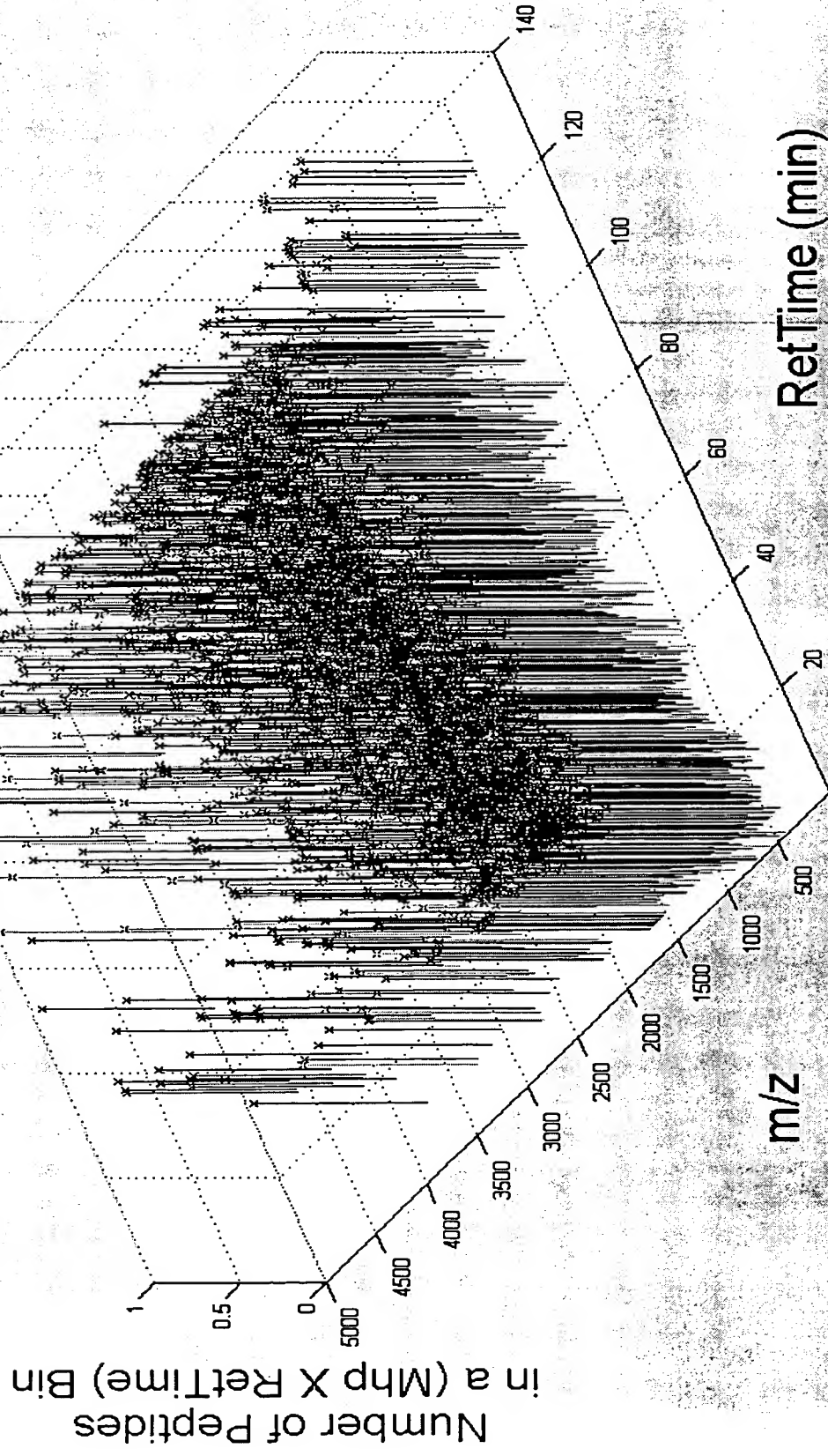
Key point: 1 peptide in a bin

Two injections

Input: mhp, retTime, Intensity, charge

for each peptide and each injection.

retTime tracking:

Injection 1: Mhp_1, retTime_1, Intensity_1, charge_1

Pair

Track

Compare

Pair

Injection 2: Mhp_2, retTime_2, Intensity_2, charge_2

Two peptides "track" if

delta m/z < 0.02 amu

delta retTime +/- 5 minutes.

delta Charge < 0.5

retTime backbone

Between two injections

– A molecule that elutes at retTime t in injection_1

– Will elute at (t + deltaT) in injection_2

– **deltaT vs T is the backbone**

– How do we find it?

By exact mass matching.

Ret time backbone: Ideal

delta retTime
Between
pair of peptides

backbone

0

retTime

- Pair of Peptides between two injections

21

retTime backbone: Real

delta retTime
Between
pair of peptides

backbone

retTime

- Pair of Peptides between two injections

22

Miracle

- With thousands of masses to choose from

  Most that satisfy m/z matching, define the retention time backbone.

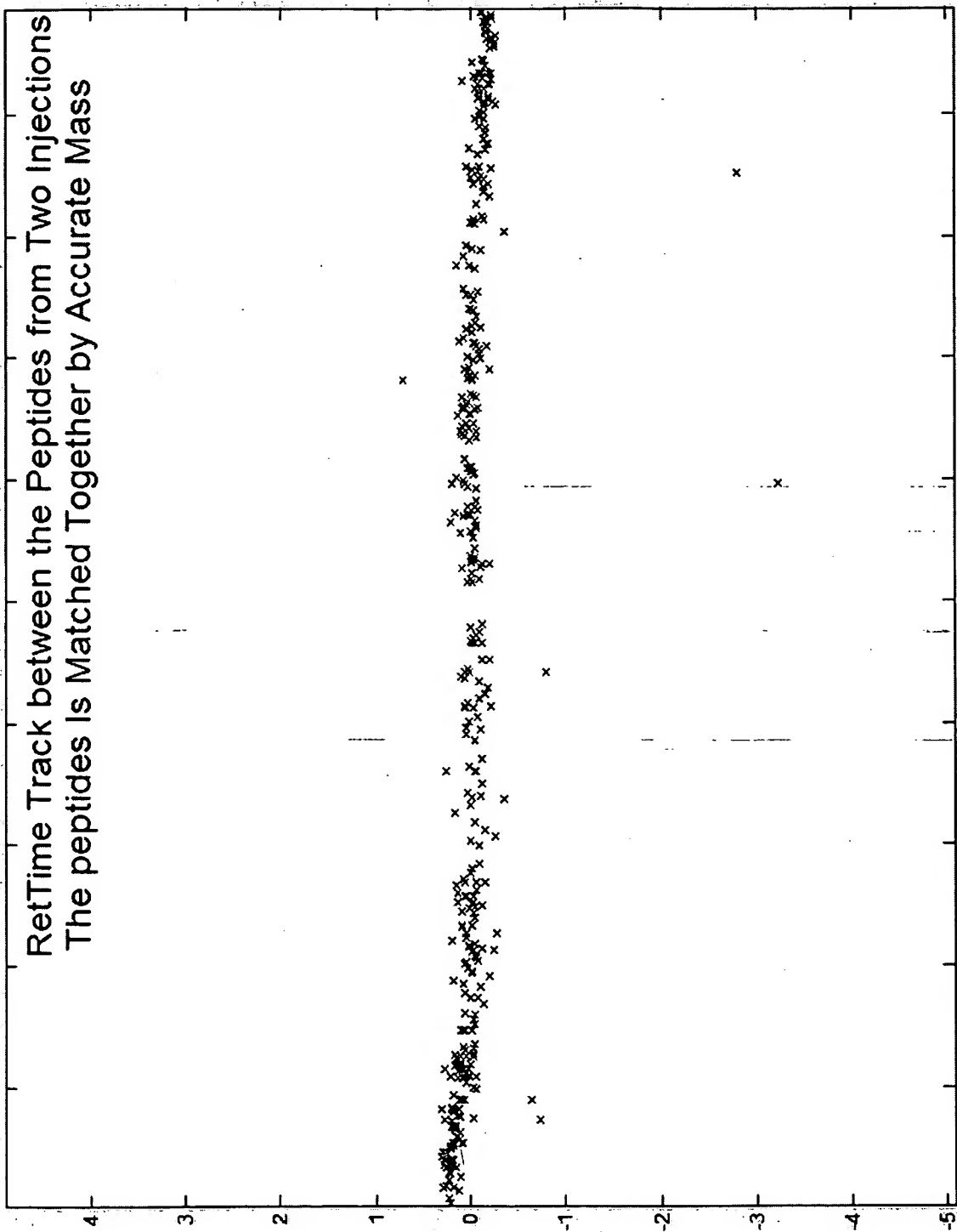 Next plot shows deltaRetTime vs retTime for 7000 points

- Most lie on back bone

  (Zoom in)
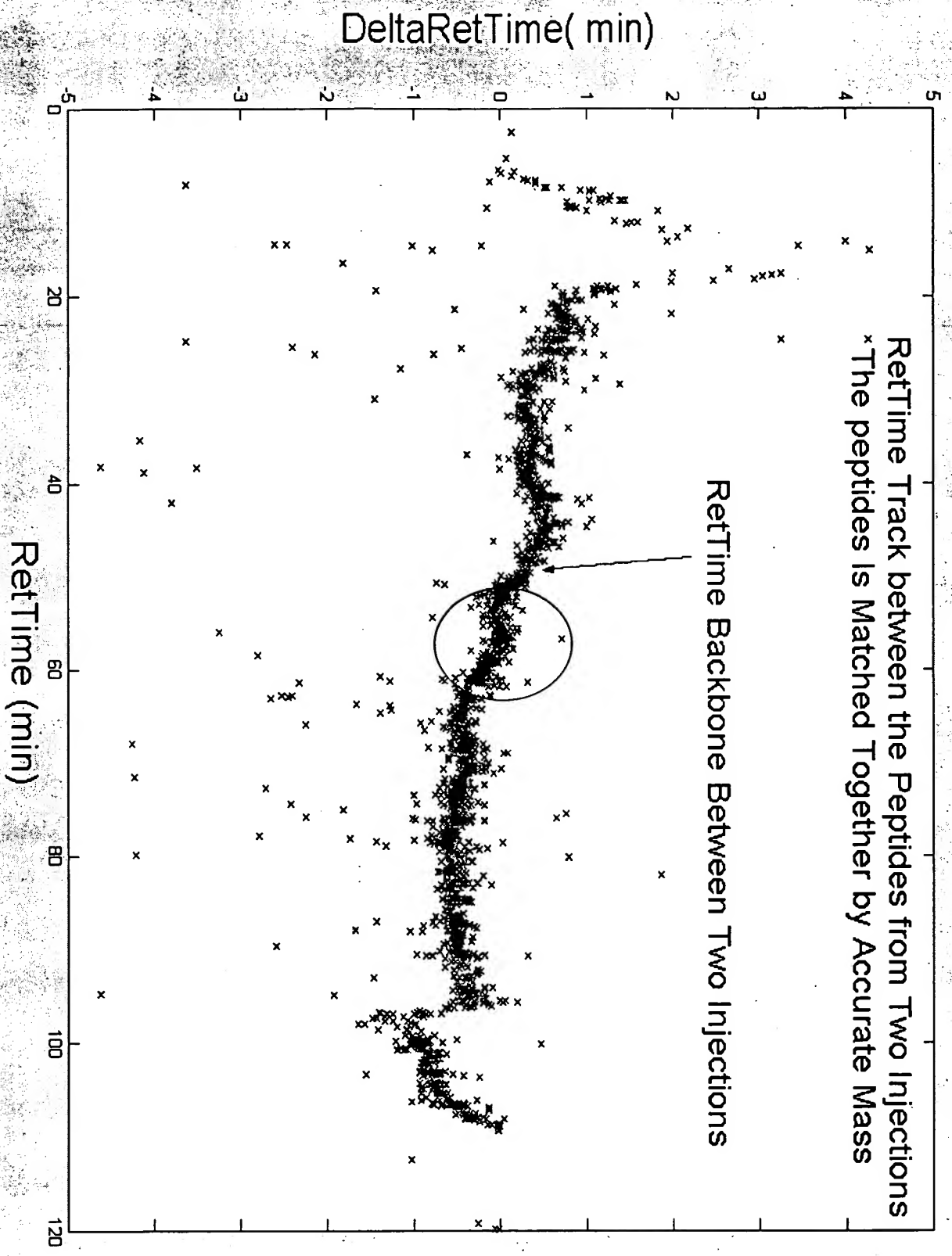
RetTime Track between the Peptides from Two Injections
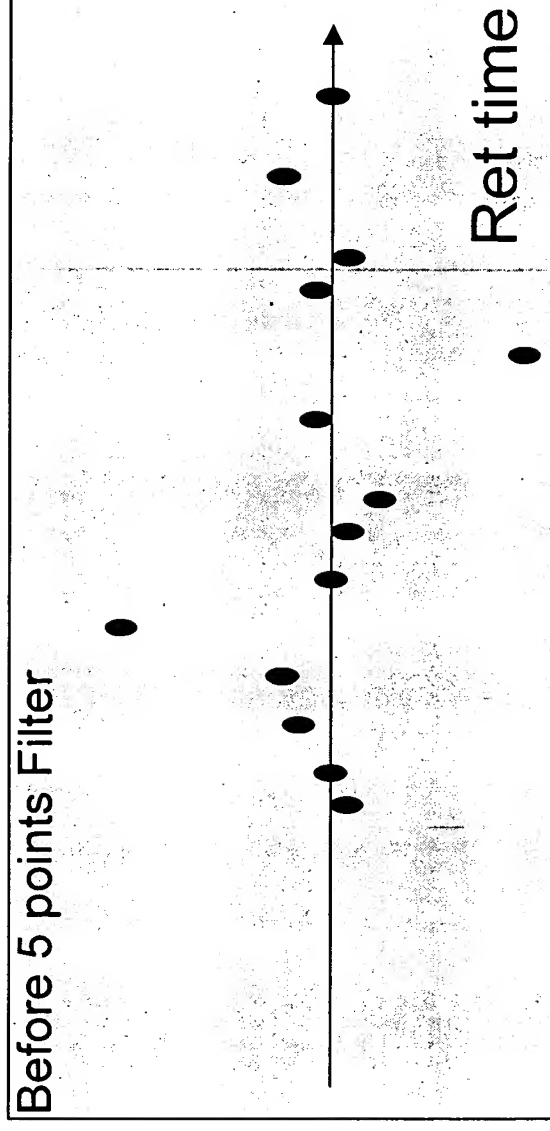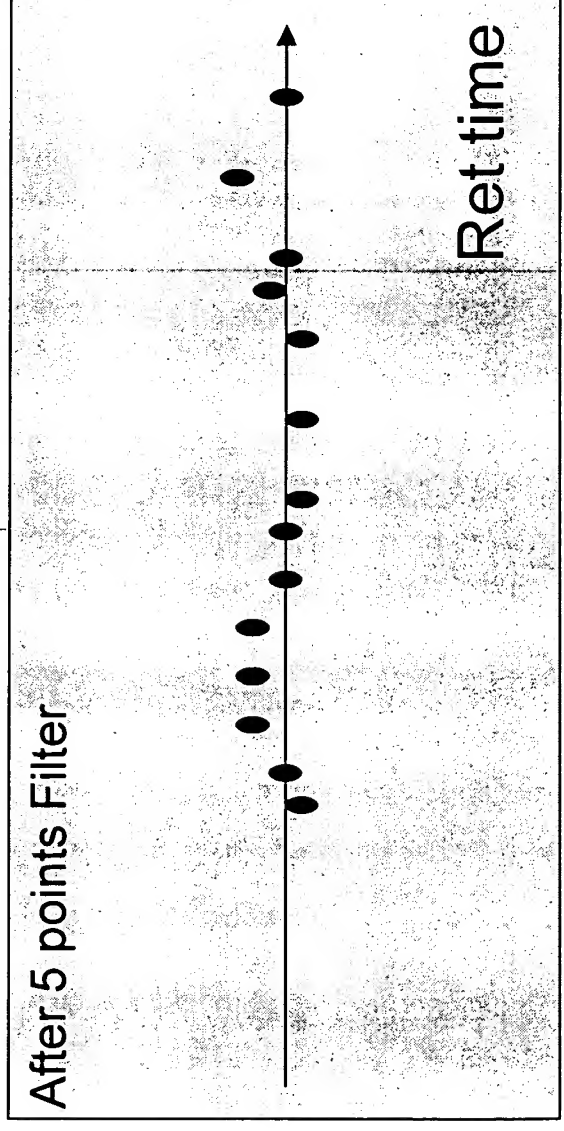The peptides Is Matched Together by Accurate Mass

RetTime Backbone Between Two Injections

RetTime Track between the Peptides from Two Injections
The peptides Is Matched Together by Accurate Mass
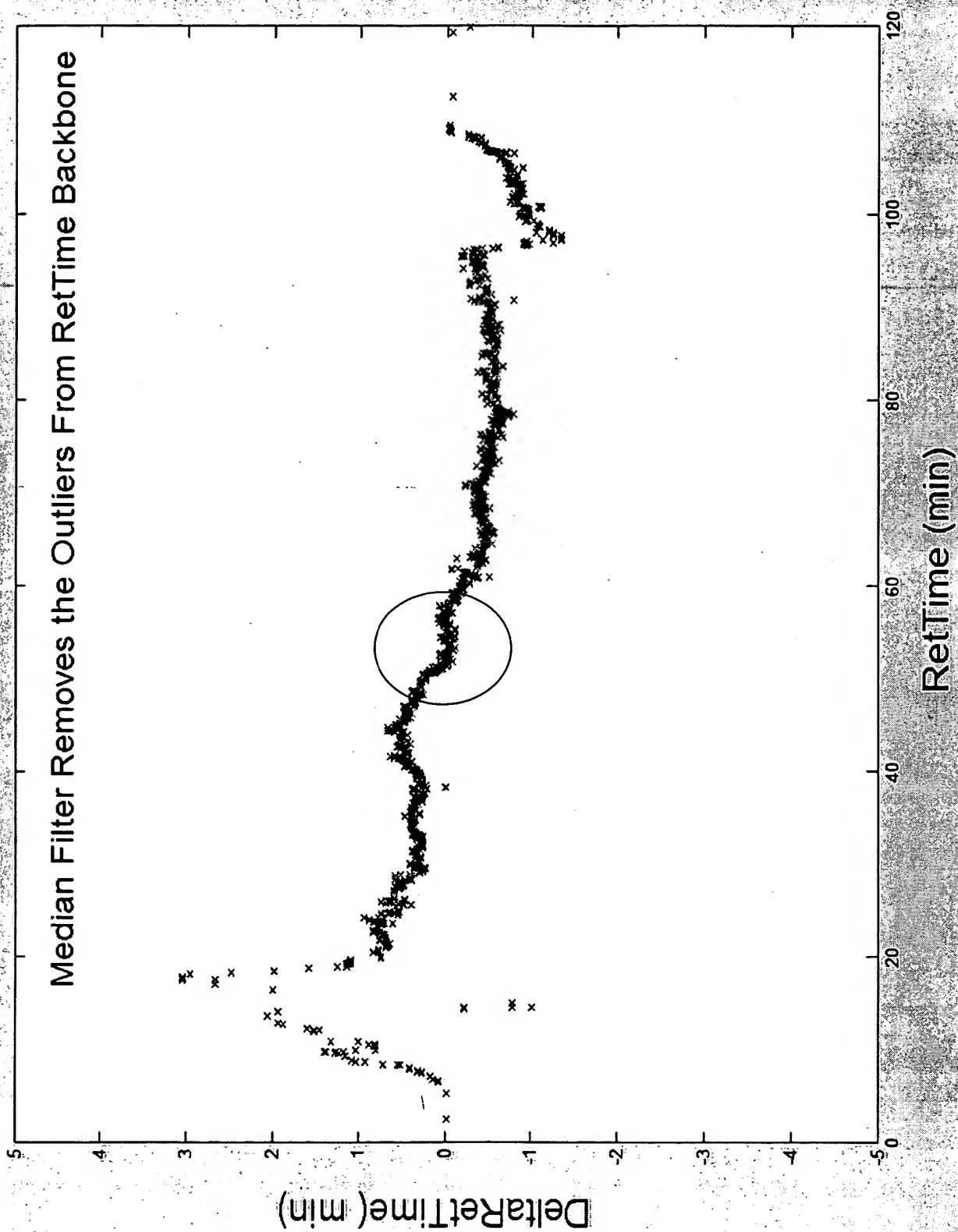
Median Filter to Remove Outliers

delta retTime
Between
pair of peptides

Before 5 points Filter

Ret time

delta retTime
Between
pair of peptides

After 5 points Filter

Ret time

26

Median Filter Removes the Outliers From RetTime Backbone

DeltaRetTime(min)

RetTime (min)

Median Filter Removes the Outliers From RetTime Backbone

Build Match Criteria Along the Backbone

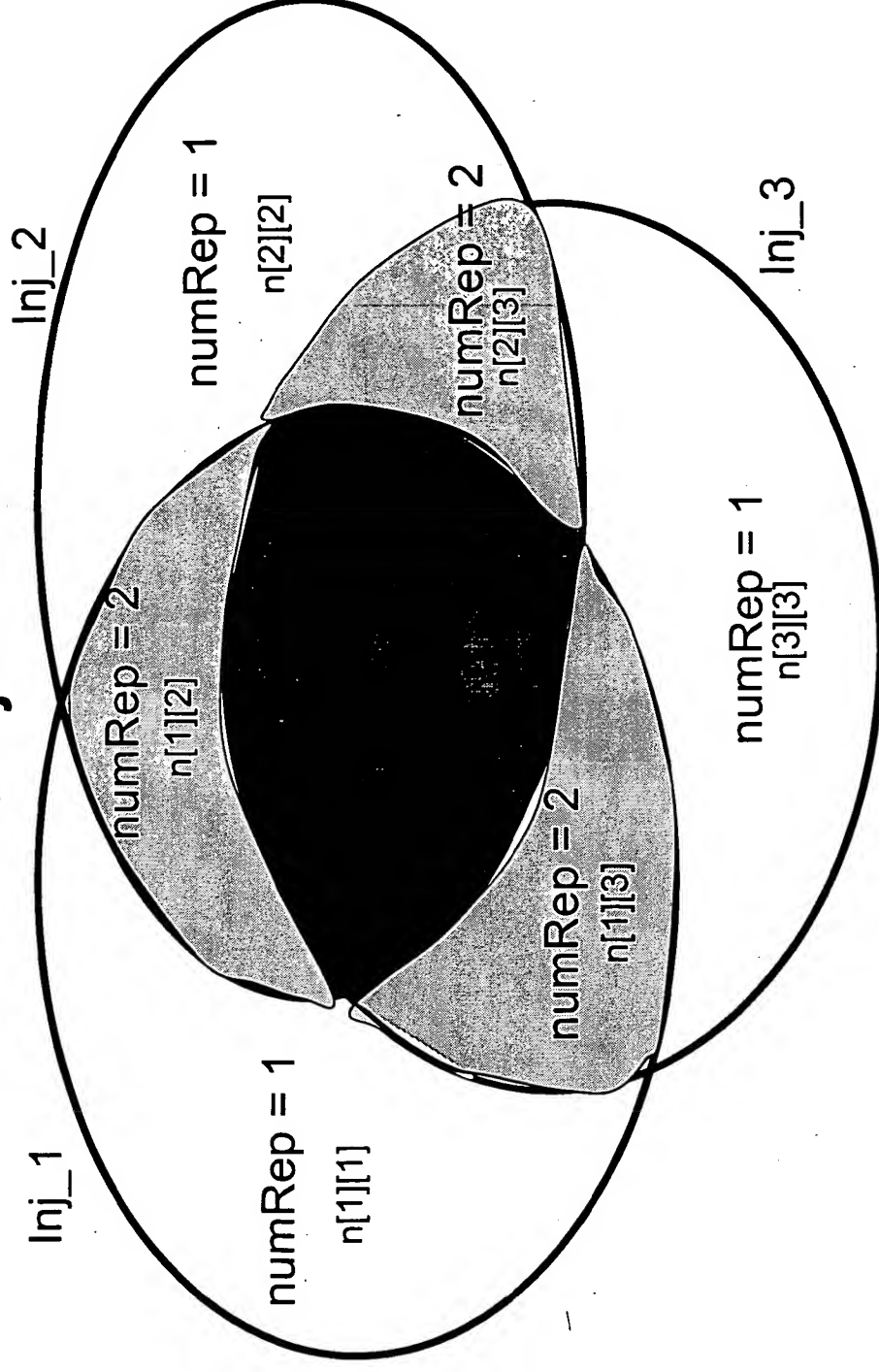Build Match Criteria Along the Backbone
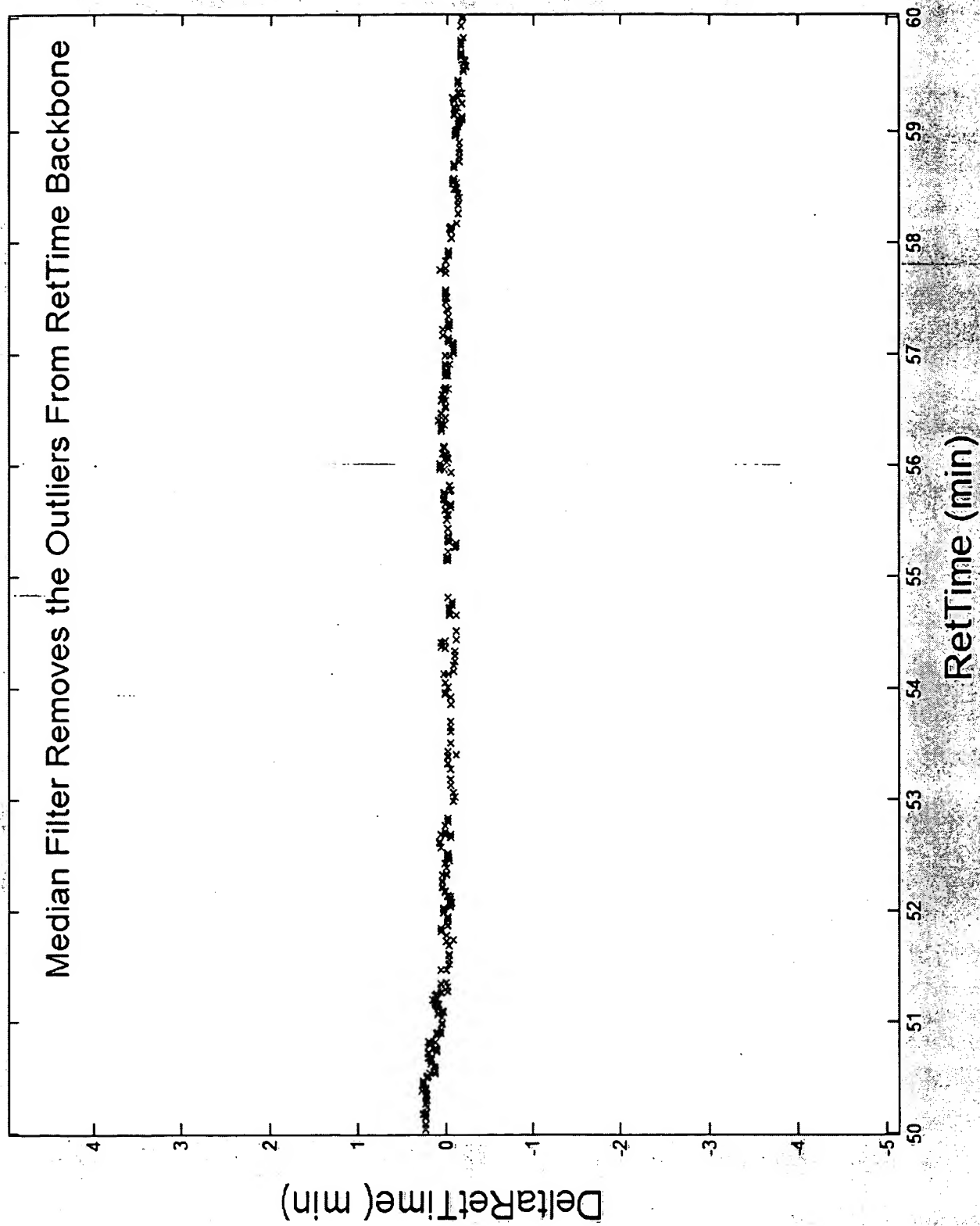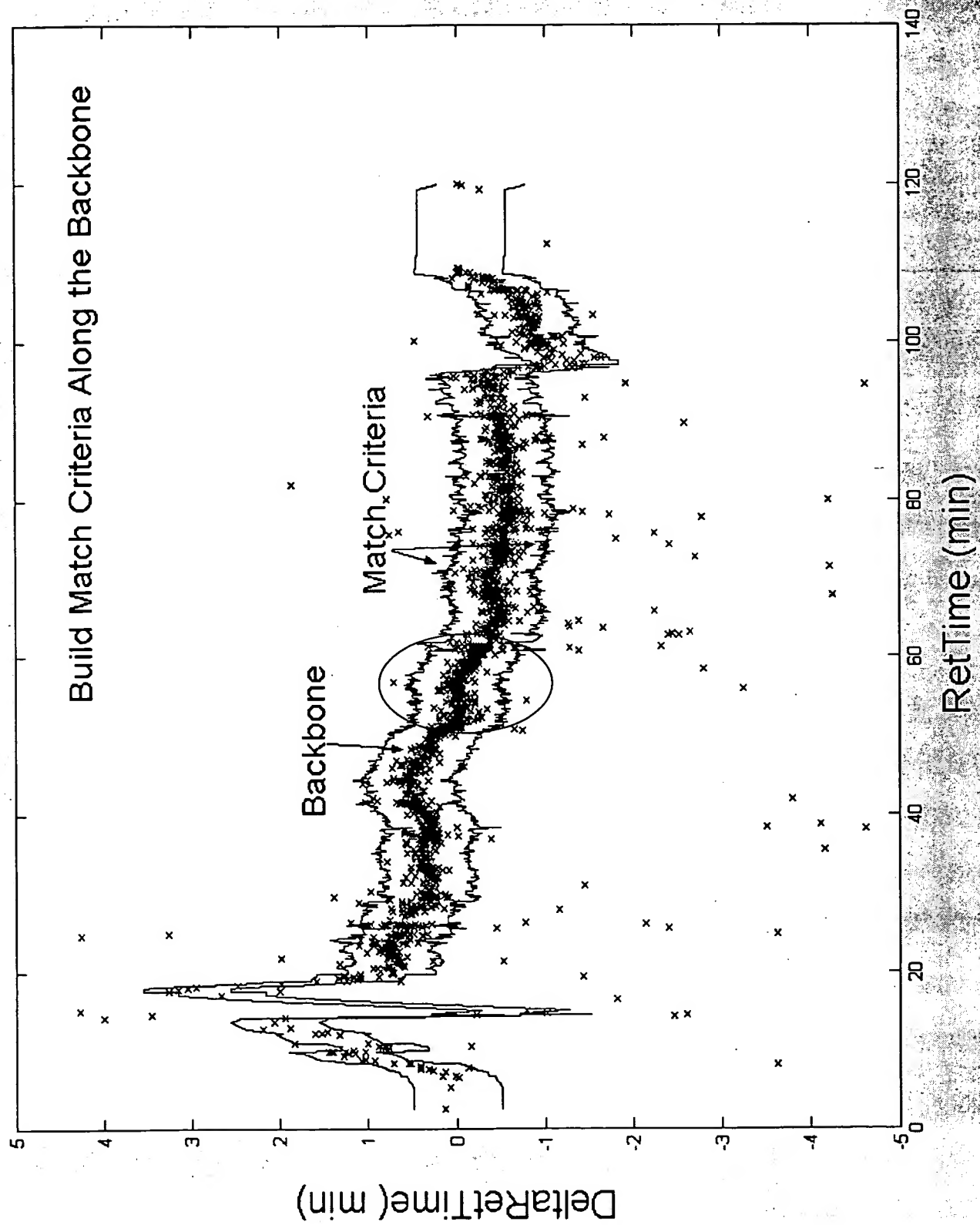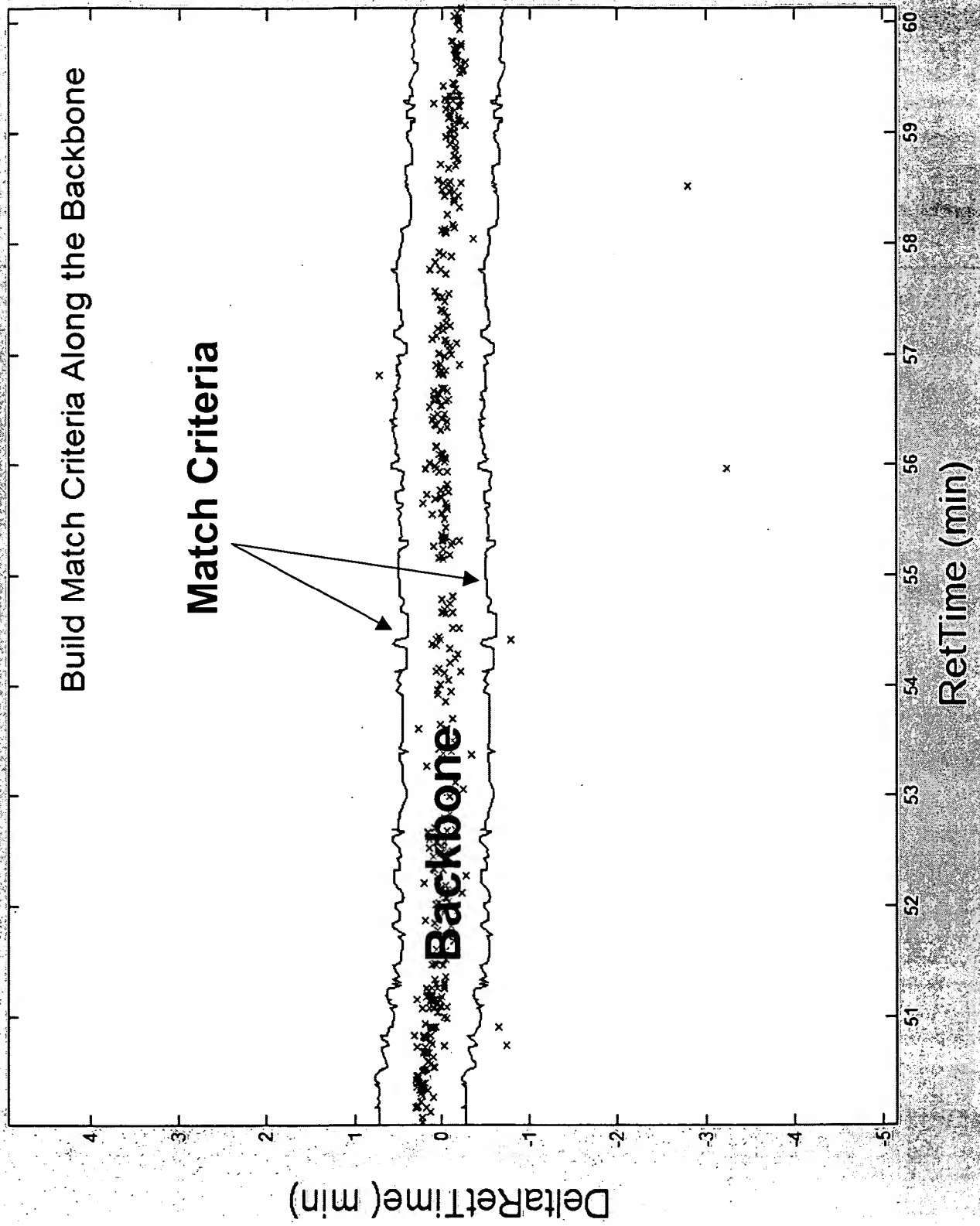
One Sample: Three Injections

Pass A: Find RetTime backbone, modify retTime

(Calibration of retTime based on accurate mass match),

Pass B: Check the symmetry between injections,

Pass C: Track peptides (or ions) among injections

Three Injections

Inj_1

Inj_2

Inj_3

numRep = 1
n[1][1]

numRep = 2
n[1][2]

numRep = 1
n[2][2]

numRep = 2
n[2][3]

numRep = 2
n[1][3]

numRep = 1
n[3][3]

$n[i][i]$: number of unmatched peptides in injection (i)

$n[i][j]$ (i != j): number of matched peptides between two injections (i and j)

$n[i][j] = n[j][i]$ when (i != j),

$n[1][2][3]$: number of matched peptides among three injections (1, 2, and 3)

	Inj = 1	Inj = 2	Inj = 3
Inj = 1	n[1][1]	n[1][2]	n[1][3]
Inj = 2	n[2][1]	n[2][2]	n[2][3]
Inj = 3	n[3][1]	n[3][2]	n[3][3]

N_T: Number of total peptides in all injections,

N [i]: Number of peptides in each injection (i = 1,2,3),

N_Rep[r]: Number of total peptides in replication (r = 1,2,3),

N[1] = n[1][1] + n[1][2] + n[1][3] + n[1][2][3],
N[2] = n[1][2] + n[2][2] + n[2][3] + n[1][2][3],
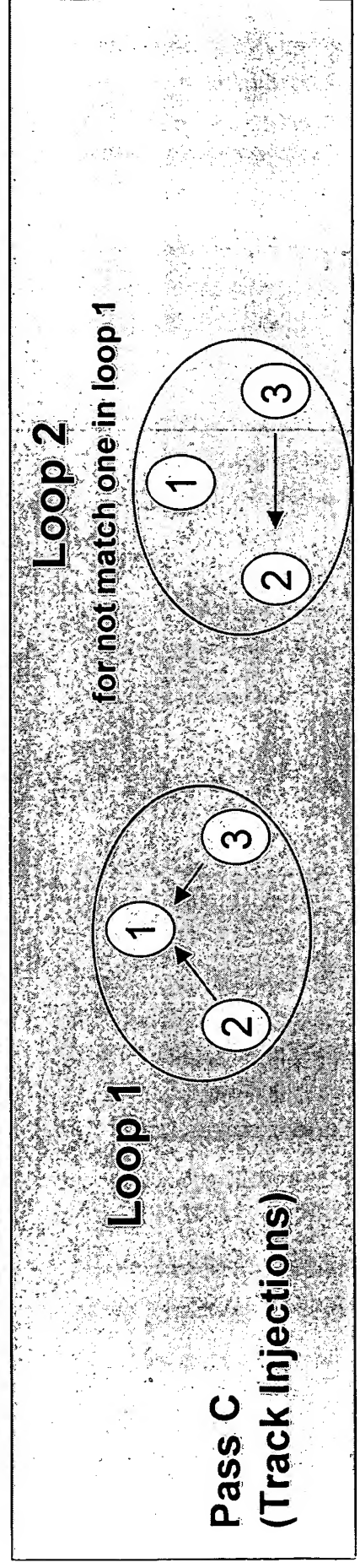N[3] = n[1][3] + n[2][3] + n[3][3] + n[1][2][3],

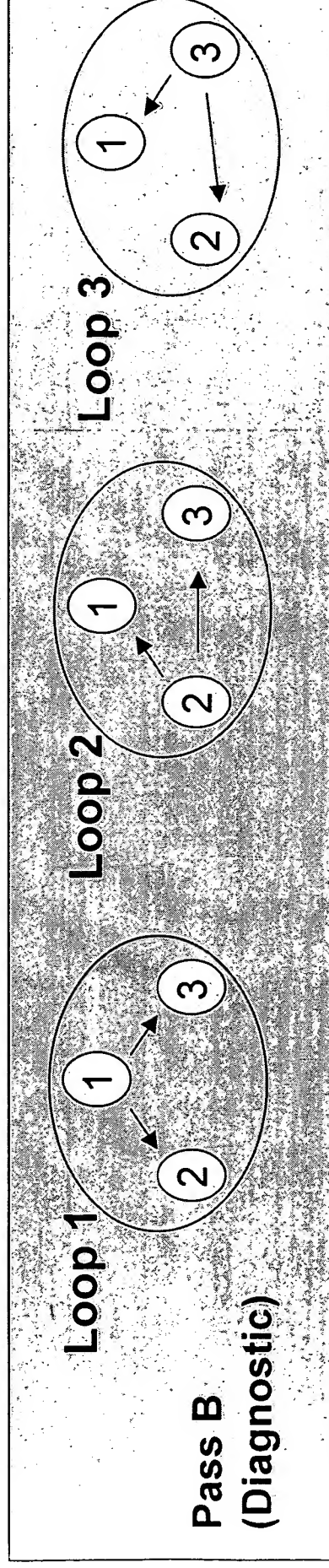N_Rep[1] = n[1][1] + n[2][2] + n[3][3],
N_Rep[2] = n[1][2] + n[1][3] + n[2][3],
N_Rep[3] = n[1][2][3],

N_T   = N[1] + N[2] + N[3]
= (n[1][1] + n[2][2] + n[3][3]) + 2*(n[1][2] + n[1][3] + n[2][3]) + 3*(n[1][2][3])

= N_Rep[1] + 2* N_Rep[2] + 3* N_Rep[3].

Three Injections Loop

Pass A
(RetTime Backbone)

Loop 1

Injection Index

Loop 2

Pass B
(Diagnostic)

Loop 1

Loop 2

Loop 3

Pass C
(Track Injections)

Loop 1

Loop 2

for not match one in loop 1

Build Match Criteria Along the Backbone

Build Match Criteria Along the Backbone

Histogram of DeltaMhp
for All Matched Peptides

Number of Peptides in a Bin

deltaMhp/Average(mhp)

x 10⁻⁶

Histogram of deltaRefRetTime for All Matched Peptides

Number of Peptides in a Bin

deltaRefRetTime/Average(refRetTime)

Histogram of deltaInt
for All Matched Peptides

deltaInt/Average(Int)

Number of Peptides in a Bin

3 Replication of 3 Injections Map
With Charge > 1

mhp

refRetTime (min)

3 Replication of 3 Injections Map.
With Charge >= 1

3 Replication of 3 Injections Map
With Charge >= 1

mhp

refRetTime (min)

3 Replication of 3 Injections Map
With Charge >= 1

mhp

refRetTime (min)

3 Replication of 3 Injections
With Charge > 1

3 Replication of 3 Injections
With Charge > 1
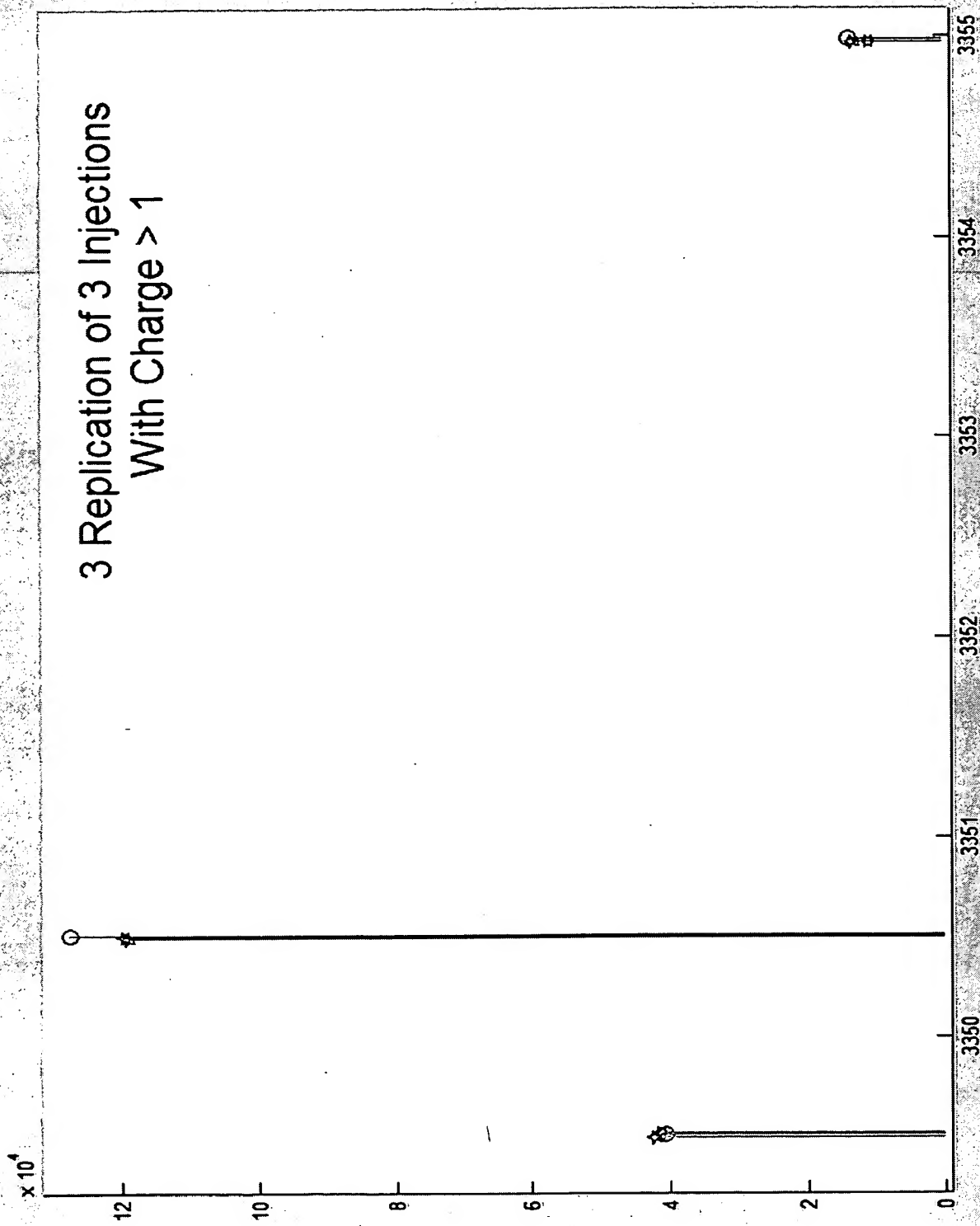
Number of Peptides and Intensity of Each Injection

Injection	Number of Peptides	Intensity
1	7273	6.63e+007
2	7336	6.89e+007
3	8497	6.59e+007
Total	23106	2.01e+008
Average	7702	6.70e+007
CV = SD/Average	8.95%	2.46%

Number of Peptides and Intensity of Each Replication

Replication X out of 3	Number of Peptides	% of Total Peptides	Peptide Intensity	% of Total Peptide Intensity
3	3 X 4356	56.56%	1.84e+08	91.52%
2	2 X 1984	17.17%	8.57e+06	4.27%
1	1 X 6070	26.27%	8.47e+06	4.21%,
Total	23106	100%	2.01e+08	100%

CV_I_Ave indicates how well the injection is replicated

$I[m][i]$: Intensity for match index $= m$ $(0 \ to \ N_m - 1)$ and injection index $= i$ $(0 \ to \ N_{inj} - 1)$

$I_{ave}[m] = \dfrac{1}{N_{inj}} \displaystyle\sum_{i=0}^{N_{inj}-1} I[m][i]$: average intensity from all replicated injections.

$sDevI[m] = \sqrt{\dfrac{1}{N_{inj}-1} \sqrt{\displaystyle\sum_{i=0}^{N_{inj}-1} \left(I[m][i] - I_{ave}[m] \right)^2}}$: standard deviation of intensity.

$CV\_I[m] = \dfrac{sDevI[m]}{I_{ave}[m]}$: coefficient of variation of intensity of replicated injections.

$CV\_I_{Ave} = \dfrac{1}{N_m} \displaystyle\sum_{i=0}^{N_m-1} CV[m]$: average coefficient of variation from all match.

Replication (X out of 3)	CV_I_Ave
3	18.07%
2	31.76%,
1	0

Two Samples: Six Injections
(Change Sample from Three Injections Case)

Sample 1: 5 pmole Gilar Proteins
(Injection_1, Injection_2, Injection_3)

Sample 2: 2.5 pmole Gilar Proteins
(Injection_4, Injection_5, Injection_6)

Six Injections Pass C

Loop 1
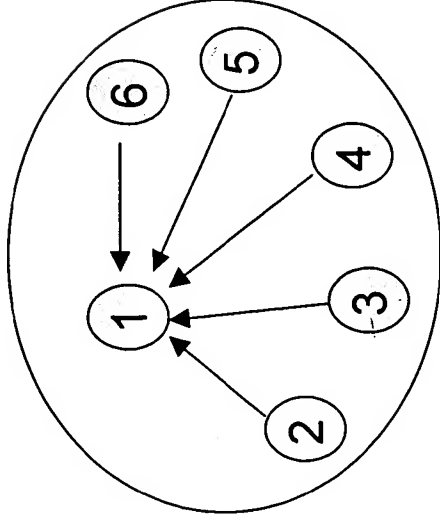
Loop 2,
for not match one in loop 1
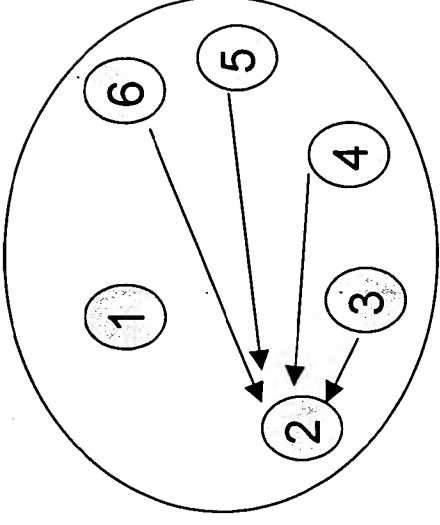
Loop 3,
for not match one in loop 1,2
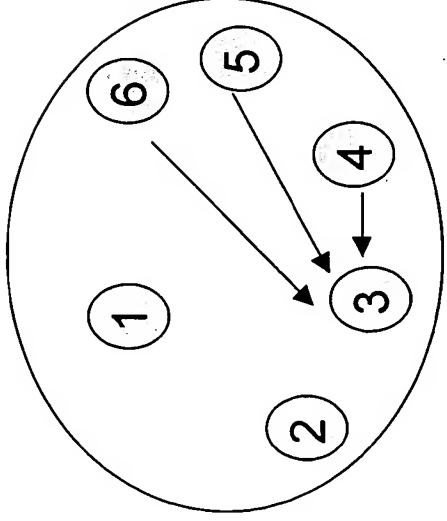
Loop 4,
for not match one in loop 1,2,3
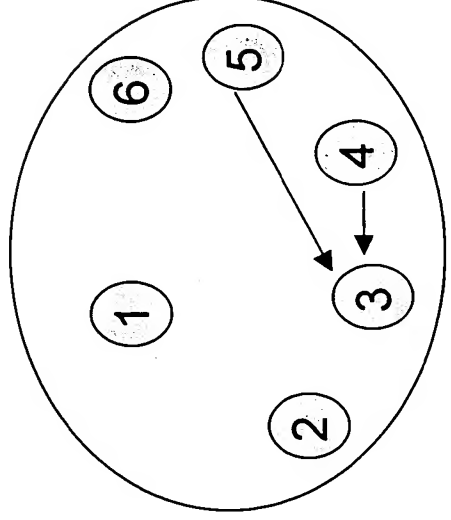
Loop 5,
for not match one in loop 1,2,3,4
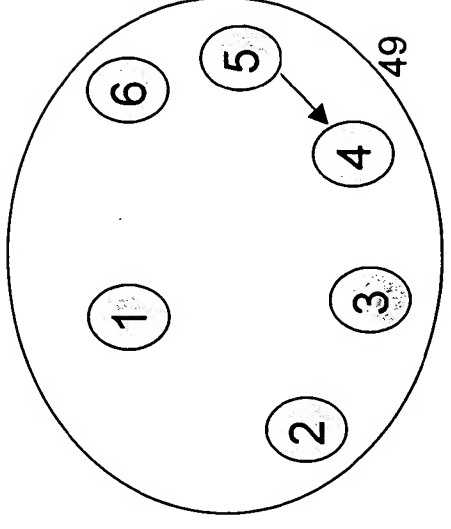
49

6 Replications of 6 Injections Match Map

mhp

retTime (min)

6 Replications of 6 Injections Match Map

mhp

retTime (min)
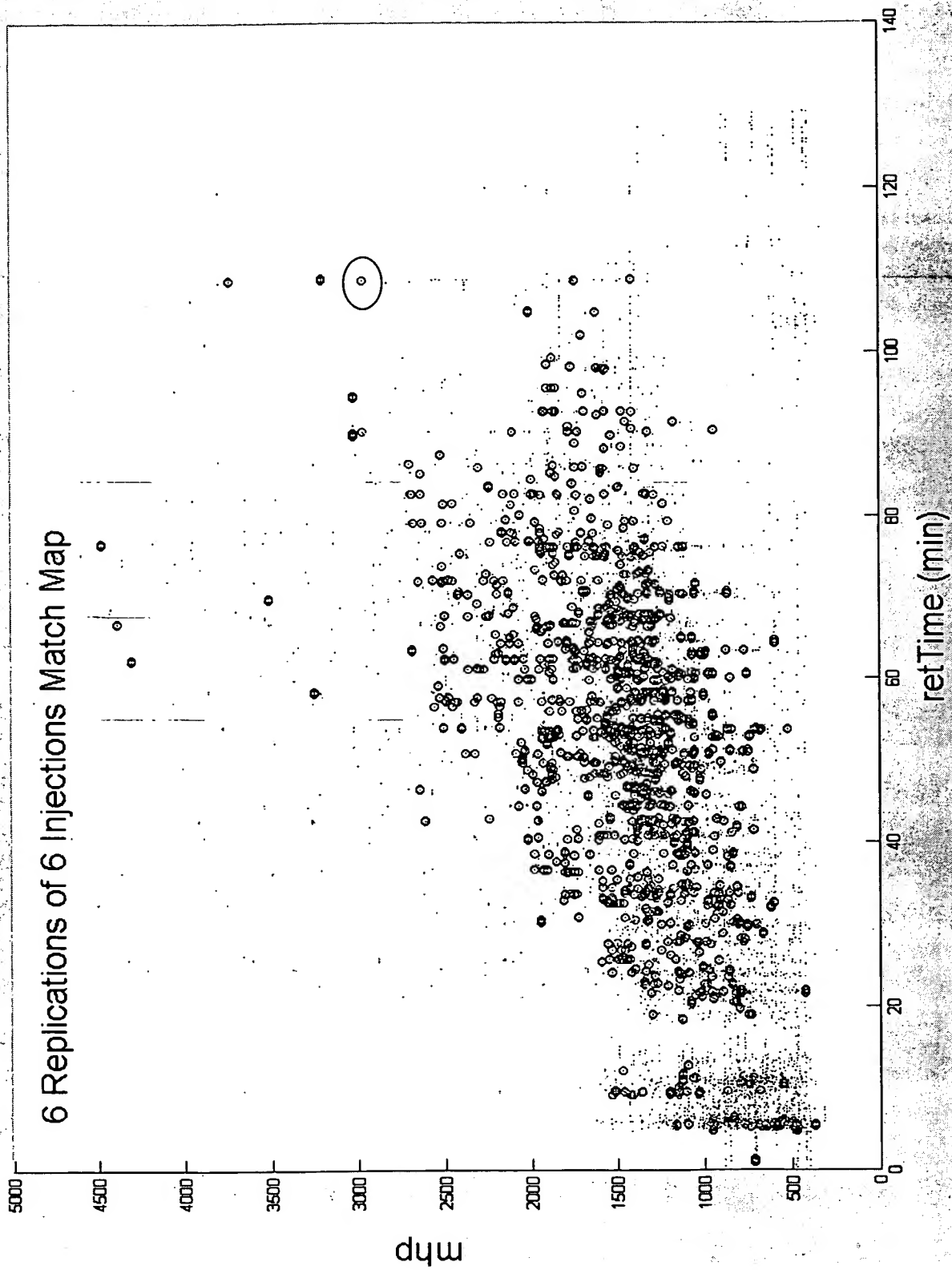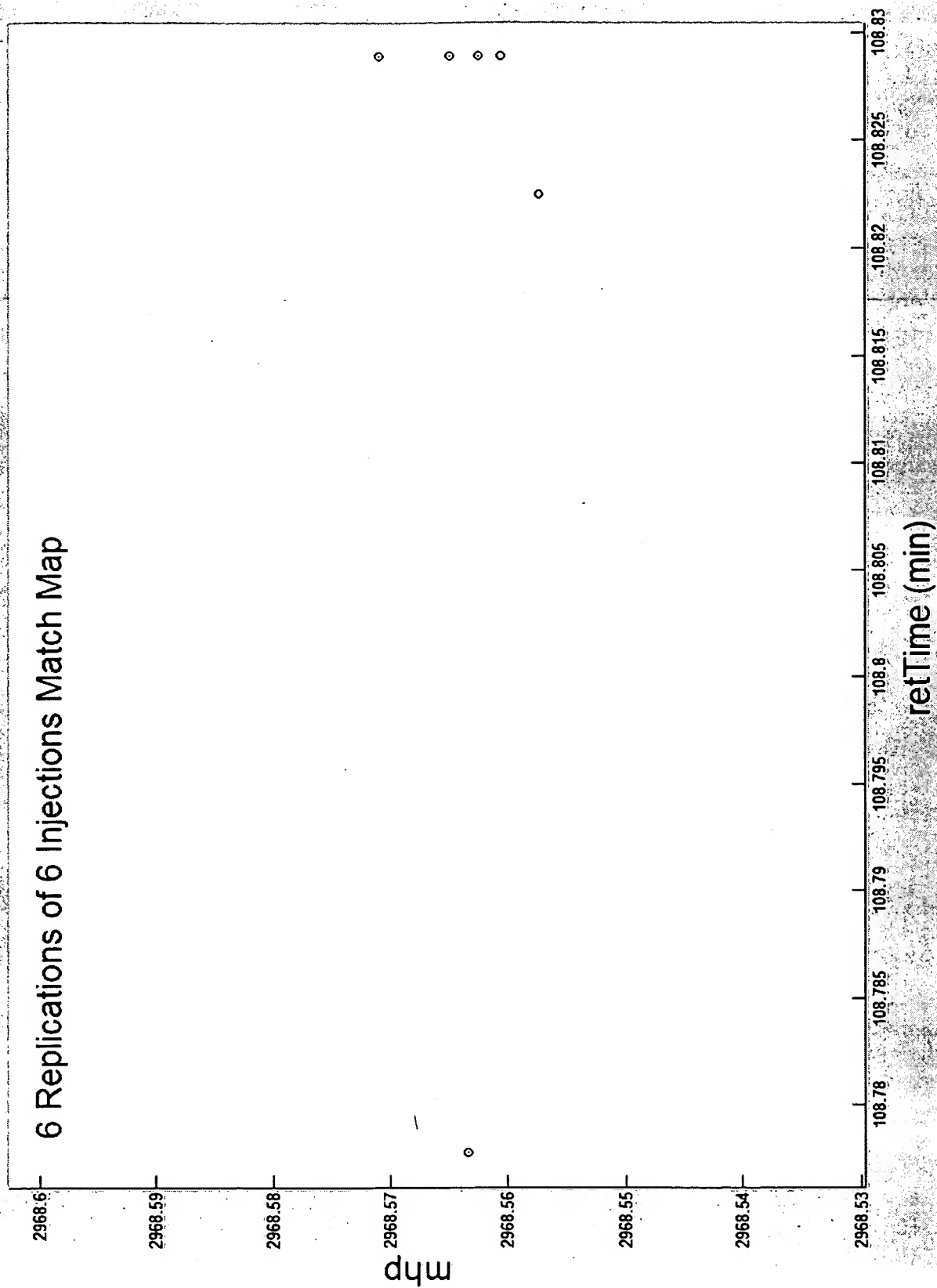
Number of Peptides and Intensity from Two Samples

Injection	Sample 1 5 pmole Gilar Number of Peptides	Sample 1 5 pmole Gilar Peptide Intensity	Sample 2 2.5 pmole Gilar Number of Peptides	Sample 2 2.5 pmole Gilar Peptide Intensity
1	4365	1.86e+07	x	x
2	4613	1.96e+07	x	x
3	4433	1.91e+07	x	x
4	x	x	2361	7.85e+06
5	x	x	2477	7.87e+06
6	x	x	2311	7.50e+06
Total	13411	5.73+07	7149	2.32+07
Average	4470	1.91+07	2383	7.74+06
CV	4.05%	3.7%	5.05%	3.8%

52

Number of Peptides and Intensity of Each Replication

Replication X out of 6	Number of Peptides	% of Total Peptides	Peptide Intensity	% of Total Peptide Intensity
6	6 X 1209	35.28%	7.237e+007	89.84%
5	5 X 366	8.90%	2.583e+006	3.21%
4	4 X 495	9.63%	1.371e+006	1.70%
3	3 X 1042	15.20%	1.861e+006	2.31%,
2	2 X 1335	12.99%	1.251e+006	1.55%
1	2 X 3702	18.01%	1.114e+006	1.38%
Total	20562	100%	2.01e+08	100%

CV_I_{Ave} (Average Coefficient Variation of Intensity from All Match in Each Sample) indicates how well the injection is replicated

	Sample 1	Sample 2
Replication (x out of 6)	CV_I_{Ave}	CV_I_{Ave}
6	11.82%	13.04%

What is intensity difference between two samples?

Intensity Ratio between Two Samples

Intensity Ratio of Two Samples
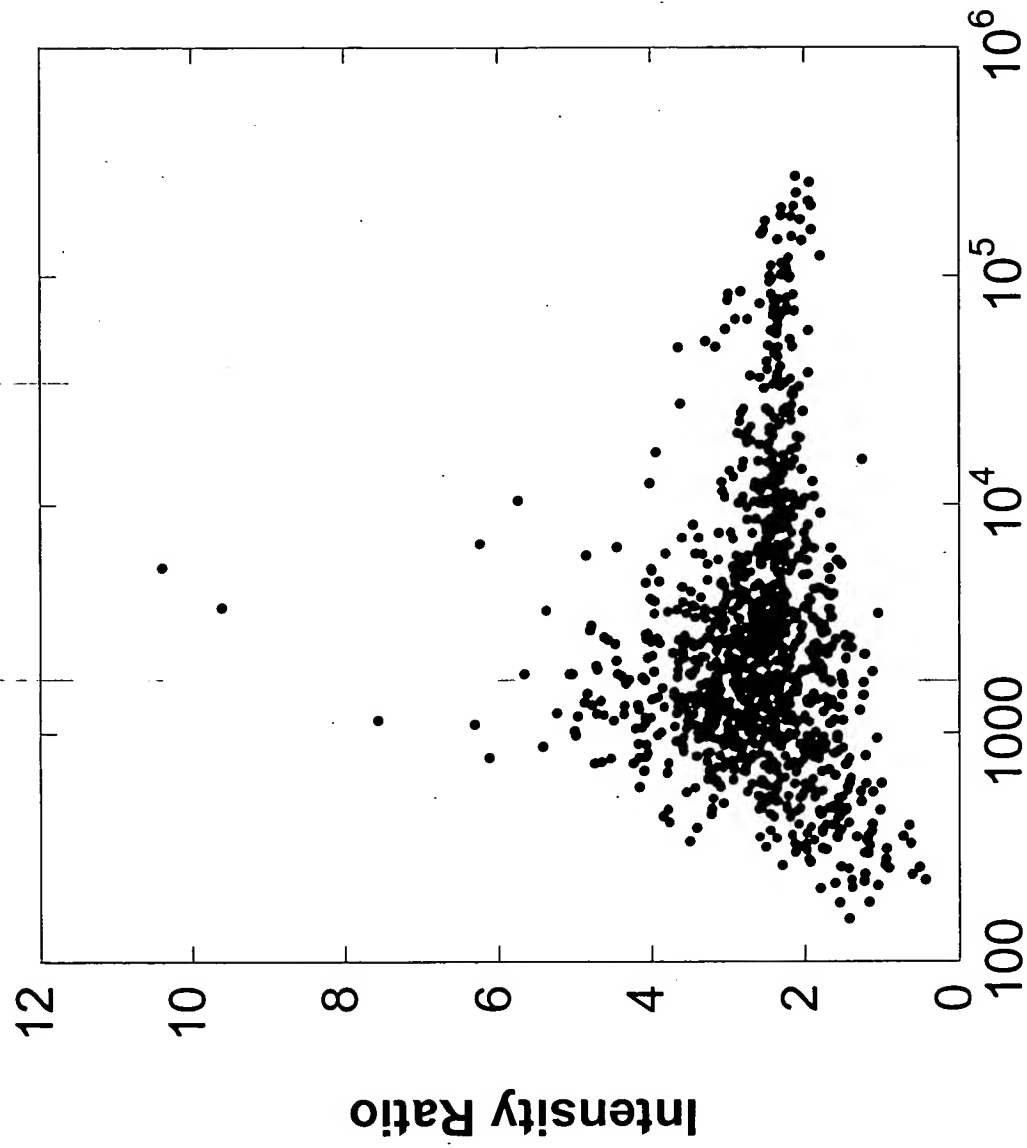
Intensity Ratio

Average Intensity of Two Samples (Log)

Intensity Ratio of Two Samples

Average CV of Two Samples (Log)

Intensity Ratio

57
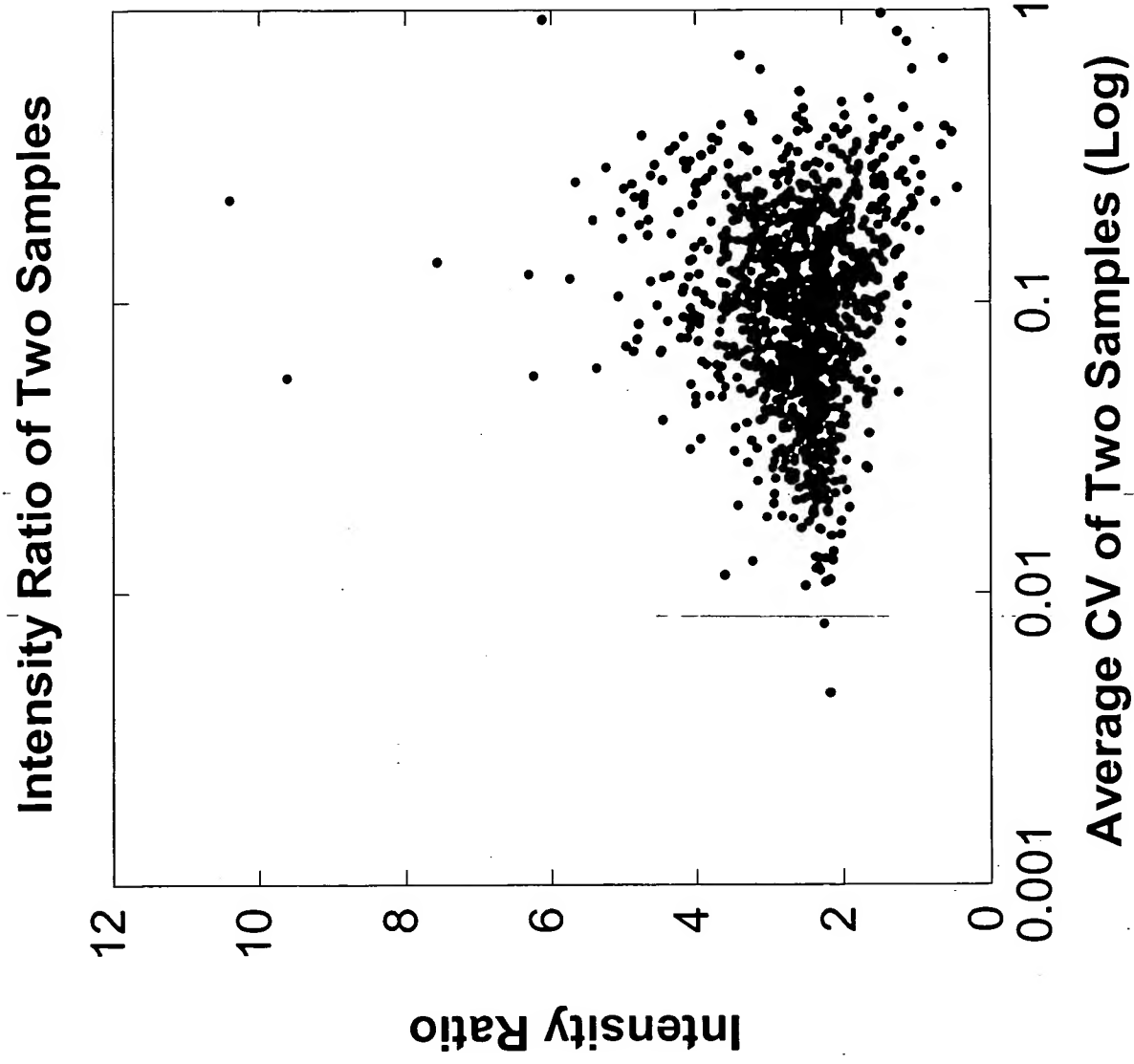
Histogram of Intensity Ratio of Two Samples

Ave_IntRatio = 2.62
Min_IntRatio = 0.46
Max_IntRatio = 10.4

Count

Intensity Ratio

58
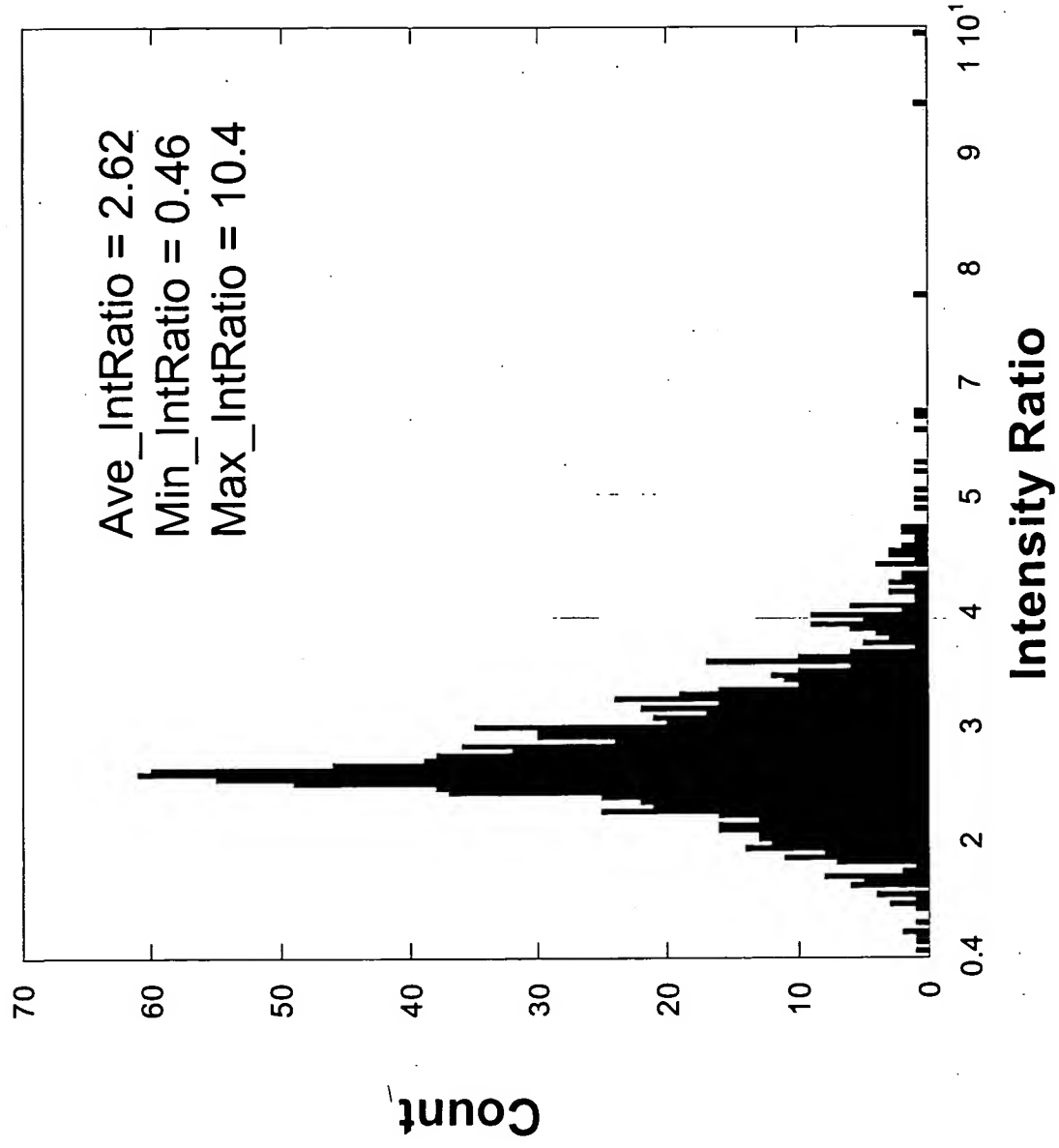
Summary

LC/MS raw data

↓

Ion sticks: m/z, retTime, Intensity

↓

Peptide sticks: mhp, retTime, Intensity

↓

Track retTime by accurate mhp

↓

Check intensity of replicated injections in one sample

↓

Compare intensity of different samples

Track3D: no limit for number of injections,
          no limit for number of samples.

```
% track three clusters

% Track 3 clusters found from runPeptideXX
% MVG June 2003

clc
delwin
clear all,pack

% load stuff
disp('*****************************************')
disp('*            Track 3 cluster            *')
disp('*            Retention time track       *')
disp('*             three peptide clusters    *')
disp('*****************************************')
disp(' ')

% load stuff


useDefault = dinput('Use default inputs',1);

if useDefault
   fileNameStkClstrA = '031403_JcsProtMix02_04Stk_01Pep';
   fileNameStkClstrB = '031403_JcsProtMix02_05Stk_01Pep';
   fileNameStkClstrC = '031403_JcsProtMix02_06Stk_01Pep';

else

    fileNameStkClstrA = uigetfile('*.mat','Pick Pep File A');
    fileNameStkClstrB = uigetfile('*.mat','Pick Pep File B');
    fileNameStkClstrC = uigetfile('*.mat','Pick Pep File C');
end

[ind0Cluster_A, indNextIso_A, numZCluster_A, numIons_A, ...
        m_zStk_A, retTimeStk_A, responseSNR_A,indPepForClstr_A] = getCluster(fileNameSt ✔
kClstrA);

[ind0Cluster_B, indNextIso_B, numZCluster_B, numIons_B, ...
        m_zStk_B, retTimeStk_B, responseSNR_B,indPepForClstr_B] = getCluster(fileNameSt ✔
kClstrB);

[ind0Cluster_C, indNextIso_C, numZCluster_C, numIons_C, ...
        m_zStk_C, retTimeStk_C, responseSNR_C,indPepForClstr_C] = getCluster(fileNameSt ✔
kClstrC);


indHitPep_B   = zeros(1,max(indPepForClstr_B));
indTestPep_B  = zeros(1,max(indPepForClstr_B));

retTimeOffset = dinput('Maximum retention time offset (min)',1);
mzOffset      = dinput('Maximum m/z offset (amu)',0.02);
stopForPlots  = dinput('Stop for all plots?',0);
stopForMissedTriple = dinput('Stop for missed Triples?',0);
SNRLimit      = dinput('SNR Limit to quit',10);

minRetTime_A = min(retTimeStk_A);
```

```
    retTimeHitB = retTimeClstr(1);
    m_zHitB     = m_zClstr(1);
    respHitB    = responseClstr(1);
    numZHitB    = numZCluster_B(indCluster);

    if (respHitB<SNRLimit)
      break
    end

    indTestPep_B(indPepForClstr_B(indCluster))=1;

    % Find hits to A
    %   save rtLimitsAB sortAB upperRTLimit lowerRTLimit
    load rtLimitsAB


    retTime0_A  = retTimeStk_A(ind0Cluster_A+1);
    m_z0_A      = m_zStk_A(ind0Cluster_A+1);
    resp0_A     = responseSNR_A(ind0Cluster_A+1);
    rtBool      = abs(retTime0_A - retTimeClstr(1)) < retTimeOffset;
    rtBoolUpper = retTime0_A - retTimeClstr(1) < interp1(sortAB,upperRTLimit,retTimeClstr ↵
(1));
    rtBoolLower = retTime0_A - retTimeClstr(1) > interp1(sortAB,lowerRTLimit,retTimeClstr ↵
(1));
    muBool      = abs(m_z0_A - m_zClstr(1)) < mzOffset;
    respBool    = abs(log10(responseClstr(1) ./ resp0_A)) < log10(2);
    zBool       = numZCluster_A == numZHitB;

    %indHit_A   = find(rtBool & muBool & respBool & zBool);
    indHit_A    = find(rtBoolUpper & rtBoolLower & muBool & respBool & zBool);

    retTimeHitA = retTimeStk_A(ind0Cluster_A(indHit_A)+1);
    m_zHitA     = m_zStk_A(ind0Cluster_A(indHit_A)+1);


    % Find hits to C
    %   save rtLimitsAB sortAB upperRTLimit lowerRTLimit
    load rtLimtsCB


    % Find hits to C
    retTime0_C  = retTimeStk_C(ind0Cluster_C+1);
    m_z0_C      = m_zStk_C(ind0Cluster_C+1);
    resp0_C     = responseSNR_C(ind0Cluster_C+1);
    rtBool      = abs(retTime0_C - retTimeClstr(1)) < retTimeOffset;
    rtBoolUpper = retTime0_C - retTimeClstr(1) < interp1(sortCB,upperRTLimit,retTimeClstr ↵
(1));
    rtBoolLower = retTime0_C - retTimeClstr(1) > interp1(sortCB,lowerRTLimit,retTimeClstr ↵
(1));
    muBool      = abs(m_z0_C - m_zClstr(1)) < mzOffset;
    respBool    = abs(log10(responseClstr(1) ./ resp0_C)) < log10(2);
    zBool       = numZCluster_C == numZHitB;

    indHit_C    = find(rtBool & muBool & respBool & zBool);
    indHit_C    = find(rtBoolUpper & rtBoolLower & muBool & respBool & zBool);
    %indHit_C   = find(rtBool & muBool & respBool );
```

61

```
[sortCB,indCB]=sort(rtCB_B);
deltaCB = deltaCB(indCB);


[sortABC_B,indSort] = sort(rtABC_B);
deltaABC_A          = rtABC_A-rtABC_B;
deltaABC_A          = deltaABC_A(indSort);


deltaABC_C          = rtABC_C-rtABC_B;
deltaABC_C          = deltaABC_C(indSort);


retTDelta = 0.2;
medianWidth = 5;
figure(3)
subplot(3,1,1)
plot(sortAB,deltaAB,'x-')

upperRTLimit = medianFilter(deltaAB,medianWidth)+retTDelta;
lowerRTLimit = medianFilter(deltaAB,medianWidth)-retTDelta;
indOverUnder= find(deltaAB > upperRTLimit | deltaAB < lowerRTLimit);

hold on
plot(sortAB,upperRTLimit,'k-')
plot(sortAB,lowerRTLimit,'k-')
plot(sortAB(indOverUnder),deltaAB(indOverUnder),'ro')
hold off
stitle('%d A Clusters hit %d B Clusters with SNR > %d', ...
[length(rtAB_B),numExamined,SNRLimit]);
ylabel('\DeltaT_{Ret} (min)')
grid on
zoom on

save rtLimitsAB sortAB upperRTLimit lowerRTLimit


subplot(3,1,2)
plot(sortCB,deltaCB,'rx-')

upperRTLimit = medianFilter(deltaCB,medianWidth)+retTDelta;
lowerRTLimit = medianFilter(deltaCB,medianWidth)-retTDelta;
indOverUnder= find(deltaCB > upperRTLimit | deltaCB < lowerRTLimit);

hold on
plot(sortCB,upperRTLimit,'k-')
plot(sortCB,lowerRTLimit,'k-')
plot(sortCB(indOverUnder),deltaCB(indOverUnder),'ro')
hold off
stitle('%d C Clusters hit %d B Clusters with SNR > %d', ...
[length(rtCB_B),numExamined,SNRLimit]);
xlabel('min')
ylabel('\DeltaT_{Ret} (min)')
grid on
zoom on

save rtLimtsCB sortCB upperRTLimit lowerRTLimit

subplot(3,1,3)
```

62

```
bar(log10(binSNR),[numAllB',numHitB'])
stitle('%d A&C Clusters hit %d B Clusters. (%d A&C Peptides hit %d B Peptides) ', ...
[length(deltaABC_A),numExamined,sum(indHitPep_B),sum(indTestPep_B)]);
xlabel('log 10 SNR')
ylabel('Number per log interval')
legend('B clusters','ABC replicates')
limits=axis;
axis([0,limits(2:4)])
grid on
zoom on

%
subplot(2,1,2)
sortRespAllB = -sort(-respAllB);
sortRespABCHit = - sort(-respABCHit(:,2));
normSum = sum(sortRespAllB);
plot(log10(sortRespAllB),100*cumsum(sortRespAllB)/normSum,'-')
hold on
plot(log10(sortRespABCHit),100*cumsum(sortRespABCHit)/normSum,'r-')
hold off
xlabel('log 10 SNR')
ylabel('Percent cumulative response')
title('Cumulative response of clusters')
legend('B clusters','ABC replicates')
grid on
zoom on

save track3save
```

63

```
% track three clusters
% Ret time Track peptides found from runPeptideXX and from Manchester's UMLG
% MVG July 2003
% Copyright © 2003 Waters Corporation
clc
delwin
clear all
pack

disp(['Executing: ',mfilename])
disp('*****************************************')
disp('*                                     *')
disp('*    Track pepetides by retention time. *')
disp('*    Inputs: 2 or more xxxPep.mat files *')
disp('*                                     *')
disp('*****************************************')
disp(' ')

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Input files -- each file is an injection
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
cdCurrent = cd;
fileNameArray=[];
pathNameArray=[];
numFile = 0;
iiInj   = 0;

fileType = {'*.mat','*.mat','*.txt'};
barMessage = {'Select xxxTrk3D.mat file','Select xxxPep3D.mat file','Select xxxCom.txt fi ↙
le'};

peptide = cell(2,13);

while 1
    dataType = menu('Select a file type, or FINISH:','Track3D','Apex3D','UMLG','FINISH');

    if dataType == 4
        break;
    end

    [filename,pathname] = uigetfile(fileType{dataType},barMessage{dataType});
    if filename==0
        continue
    else
        numFile = numFile+1;
        iiInj = iiInj+1;
        fileNameArray{numFile}=[filename];
        pathNameArray{numFile}=[pathname];

        cd(pathname)
        disp(['Process ',filename])
    end

    if dataType == 1

        [mwHPlusPep,retTimePep,responsePep,numZPep,idCluster]=Track3DGetTrk3D02(filename);
        peptide(iiInj,1:5)={mwHPlusPep,retTimePep,responsePep,numZPep,idCluster};
```

lo4

```
numZPepTarget                   = peptide{param.iiTarget,4};


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Control parameters
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

fileNameTarget             = param.fileNameArray{param.iiTarget};
param.fileNameTrack3D       = dinput('Filename of log, comp,  and .mat file',fileNameTarg ✔
et(1:(end-4)));

filenameDiary = [param.fileNameTrack3D,'log.txt'];
if exist(filenameDiary)
  delete(filenameDiary)
end
diary(filenameDiary)

disp('*****************************************')
disp('*          TrackNPeptides              *')
disp('*****************************************')
disp(mfilename)
disp(datestr(now))
disp('*****************************************')
disp('Files to be processed:')
disp(' ')
for ii = 1:param.numInjections
  disp(param.fileNameArray{ii})
end
disp(' ')
disp('*****************************************')
disp('Input parameters')
param.retTimeOffsetTracking = dinput('Retention time offset for tracking pass (min)',5.0);
param.heightRatioTracking   = dinput('Height ratio range for tracking pass ',0.4);
param.SNRLimitTracking      = dinput('SNR Limit for tracking pass',100);
param.SNRLimitID            = dinput('SNR Limit for identification',10);
param.mzOffset              = dinput('Maximum m/z offset (amu)',0.02);
param.zLowerLimit           = dinput('Lower fraction charge limit',1.5);
param.zUpperLimit           = dinput('Upper fraction charge limit',10.0);
param.retTimeThresholdID    = dinput('RetTime Threshold for ID (-1=auto)',-1);
param.retTimeStart          = dinput('Start at target retention time (min)',min(retTimeTa ✔
rget));
param.retTimeEnd            = dinput('End at target retention time (min)',max(retTimeTarg ✔
et));
param.zThreshold = 0.5;

% Definition
%     param.sigmaRetTimeResid(ii) = median(abs(retTimeResidNoZero))/0.67;


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Normalize responses using median
% of peptides that hit
%
% Construct retention time map w/r to target
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
iiPlot = 0;
```

65

```
% Find ret time map from the ones that hit between target and A
retTimeRef          = retTimeA(indHitA);
retTimeDelta        = retTimeA(indHitA) - retTimeTarget(indHitTarget);
deltaMwHPlus        = mwHPlusA(indHitA) - mwHPlusTarget(indHitTarget);
mwHPlusHit          = mwHPlusA(indHitA);

% Sort by ret time A (why not target?)
[retTimeRef,iSrt] = sort(retTimeRef);
retTimeDelta        = retTimeDelta(iSrt);
deltaMwHPlus        = deltaMwHPlus(iSrt);
mwHPlusHit          = mwHPlusHit(iSrt);

% Need to delete the occasional coincidence in order to use interp1
indDelete = find(diff(retTimeRef)==0.0);
retTimeRef(indDelete)    =[];
retTimeDelta(indDelete) =[];
deltaMwHPlus(indDelete) =[];
mwHPlusHit(indDelete)    =[];

% Find the backbone.
retTimeDeltaMedian= medianFilter(retTimeDelta',medianWidth)';

% Obtain the residuals about the backbone
retTimeResiduals            = retTimeDelta-retTimeDeltaMedian;
retTimeResidNoZero          = retTimeResiduals(find(retTimeResiduals~=0.0));

% Determine threshold, in a round about way.
if param.retTimeThresholdID<=0.0

    param.sigmaRetTimeResid(ii) = median(abs(retTimeResidNoZero))/0.67;
else
    param.sigmaRetTimeResid(ii) = param.retTimeThresholdID/4.0;
end
% Obtain cutout
upperRTLimit        = retTimeDeltaMedian+4.0*param.sigmaRetTimeResid(ii);
lowerRTLimit        = retTimeDeltaMedian-4.0*param.sigmaRetTimeResid(ii);
indOverUnder        = find(retTimeDelta > upperRTLimit | retTimeDelta < lowerRTLimit);
indGood             = find(retTimeDelta < upperRTLimit | retTimeDelta > lowerRTLimit);

% Store cutout
peptide{ii,7}     = retTimeRef;
peptide{ii,8}     = upperRTLimit;
peptide{ii,9}     = lowerRTLimit;

% Histogram mwHPlus error
figure(5)
deltaMwHPlusGood = deltaMwHPlus(indGood);
mwHPlusHitGood    = mwHPlusHit(indGood);

ppmHit            = 1e6*deltaMwHPlusGood./mwHPlusHitGood;

binVec = min(deltaMwHPlusGood):0.001:max(deltaMwHPlusGood);
binPpm = min(ppmHit):0.5:max(ppmHit);

subplot(param.numInjections-1,2,2*iiPlot-1)
hist(deltaMwHPlusGood,binVec)
```

1.6

```matlab
        title(sprintf('Histogram about %d pt median. Inj %d - target. %d zeros removed. Sigma
    = %6.3f min',...
        medianWidth,ii,length(retTimeResiduals==0.0),param.sigmaRetTimeResid(ii)))
        xlabel('min')
        ylabel('points per 0.01 min')
        grid on
end

% move windows a bit for a better view.
screenSize      = get(0,'screensize');
screenHalf      = screenSize(3)/2;
screenHeight    = screenSize(4);

set(3,'position',[1 1              screenHalf screenHeight/2])
set(4,'position',[screenHalf 1 screenHalf screenHeight/2])

drawnow


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%% Call pass B of comparison program  %%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

peptide = track3DPassB02 (peptide,param);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%    Plot  ret time maps,
%    and histogram of delta ret time
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
iiPlot = 0;

for ii = 1:param.numInjections

    % No need to change response of target.
    % No need to construct ret time map of target
    if (ii==param.iiTarget)
        continue;
    end

    iiPlot=iiPlot+1;


    responseA           = peptide{ii,3};
    retTimeA            = peptide{ii,2};
    mwHPlusA            = peptide{ii,1};

    indHit              = peptide{ii,11};
    indHitA             = indHit(:,1);
    indHitTarget        = indHit(:,2);

    % Find ret time map
    retTimeHitA         = retTimeA(indHitA);
    retTimeDelta        = retTimeA(indHitA) - retTimeTarget(indHitTarget);

    [retTimeHitA,iSrt]= sort(retTimeHitA);
    retTimeDelta        = retTimeDelta(iSrt);
```

67

```
        logRespRatio            = log10(responseA(indHitA)./responseTarget(indHitTarget));
        sigmaRespRatioLog       = median(abs(logRespRatio))/0.67;
        hist(logRespRatio,100)
        xlabel('log10(ratio)')
        ylabel('Number per interval')
        stitle('Std dev  respRatios for Inj %d = %5.3f',[ii,10.0.^sigmaRespRatioLog])
        grid on
        zoom on

end


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%    Find all N peptides that replicate
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


% indHitMatAll(ii,jj) = index of peptide in injection jj that hit target peptide ii
indHitMatAll = zeros(length(responseTarget),param.numInjections);
for ii = 1:param.numInjections

    if (ii==param.iiTarget)
        continue
    end

    indHit              = peptide{ii,11};
    indHitA             = indHit(:,1);
    indHitTarget        = indHit(:,2);

    indHitMatAll(indHitTarget,ii) = indHitA;

end


% help logical
% LOGICAL Convert numeric values to logical.
%    ...
%    The term "logical indexing" refers to any indexing operation where
%    the index expression is a logical array, in which case the index is
%    treated as a mask that selects elements from the indexed array.  In
%    essence, it is a short-hand notation for A(FIND(B)) that enables us
%    to simply write A(B) when B is a logical array.  The result is the
%    elements of A at the indices where B is one.  It is often convenient
%    to derive the index expression from the indexed data itself.  For
%    example, the positive elements of a vector A can be obtained using
%    A(A>0).

indHitMatAllOmit   = indHitMatAll;
indHitMatAllOmit(:,param.iiTarget)=[];

indHitLogical = indHitMatAllOmit>0;

if param.numInjections ==2
    indHitSum       = indHitLogical;
else
    indHitSum       = sum(indHitLogical')';
end

hitVecLogical = (indHitSum >= (param.numInjections-1));
totalHits = sum(hitVecLogical)
```

```
numPerBinHitTarget=[];
numTotalHitN{1} = int2str(length(indZTarget));
for ii = 1:(param.numInjections-1)
  hitVecNLogical = (indHitSum >= ii);
  numPerBinHitTarget(ii,:) = hist(responseTarget(find(hitVecNLogical)),binSNR);
  numTotalHitN{ii+1} = int2str(sum(hitVecNLogical));
end

bar(log10(binSNR),[numPerBinTarget',numPerBinHitTarget'])

stitle('%d Peptides hit %d Target peptides', [totalHits,length(indZTarget)]);
xlabel(['SNR (log10). Algorithm: ',param.inputDataAlgorithm,'. Target: ',fileNameTarget,'
. Log File: ',param.fileNameTrack3D])
ylabel('Number per log interval')
legend(numTotalHitN)
limits=axis;
axis([logLimitLower,logLimitUpper, -10, limits(4)])
grid on
zoom on


subplot(2,1,2)
responseTargetSort = -sort(-responseTarget(indZTarget));
normSum = sum(responseTarget(indZTarget));
plot(log10(responseTargetSort),100*cumsum(responseTargetSort)/normSum,'-')

for ii = 1:(param.numInjections-1)
    hitVecNLogical = (indHitSum >= ii);
    responseTargetHit = responseTarget(find(hitVecNLogical));
    responseTargetHitSort = -sort(-responseTargetHit);
    hold on
    plot(log10(responseTargetHitSort),100*cumsum(responseTargetHitSort)/normSum,'r-')
    hold off
end
xlabel('log 10 SNR')
ylabel('Percent cumulative response')
title('Cumulative response of hits')
legend(param.fileNameArray)
limits=axis;
axis([logLimitLower,logLimitUpper, -5, 105])
grid on
zoom on

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Write text file output
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
compareFileName = [param.fileNameTrack3D,'Comp.txt'];
if exist(compareFileName)==2
  delete(compareFileName)
end
disp(['Write comparefile:  ' compareFileName])
fid = fopen(compareFileName,'w');
fprintf(fid,' ID  | mwHPlus  |retTime |response |fracZ');

for jj = 1:param.numInjections
    ii jj==param.iiTarget
```

```
end
disp('****************************************')

disp(['Finished executing: ',mfilename])
diary off
```

7i

```matlab
peptide = track3DPassA02 (peptide,param);
for ii = 1:param.numInjections

    % Find ret time map from the ones that hit between target and A
    retTimeRef       = retTimeA(indHitA);
    retTimeDelta     = retTimeA(indHitA) - retTimeTarget(indHitTarget);

    % Sort by ret time A
    [retTimeRef,iSrt] = sort(retTimeRef);
    retTimeDelta     = retTimeDelta(iSrt);

    % Find the backbone.
    retTimeDeltaMedian= medianFilter(retTimeDelta',medianWidth)';

    % Compute reference ret time for each entity, and store it.
    retTimePepTargetA  = retTimeA - interp1(retTimeRef,retTimeDeltaMedian,retTimeA);
    peptide{ii,10}     = retTimePepTargetA;

end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function peptide = track3DPassA02 (peptide,param)
for ii = 1:length(responseTarget)

    thisRetTime  = retTimeTarget(ii);
    thisMwHPlus  = mwHPlusTarget(ii);
    thisResponse = responseTarget(ii);
    thisNumZ     = numZTarget(ii);

    for jj = 1: param.numInjections

        indSubset    = find(abs(retTimeA-thisRetTime)<param.retTimeOffsetTracking);

        mwHPlusASub    = mwHPlusA(indSubset);
        responseASub   = responseA(indSubset);
        numZASub       = numZA(indSubset);

        muBool       = abs(mwHPlusASub - thisMwHPlus)/thisNumZ < param.mzOffset;
        respBool     = abs(log10(responseASub/thisResponse)) < abs(log10(param.heightRatio ✓
Tracking));
        zBool        = abs(thisNumZ-numZASub) < param.zThreshold;

        indHitA      = indSubset(find(muBool & respBool & zBool));

        if length(indHitA)==1
                indHitA1 = peptide{jj,6};
                indHitA1 = [indHitA1;[indHitA,ii]];
                peptide{jj,6} = {indHitA1};
        end
    end
end
```

```
% [peptide] = track3DPassA01 (peptide,param);
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%    Storage map
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%   peptide{ii,1}  = mwHPlusPep;
%   peptide{ii,2}  = retTimePep,
%   peptide{ii,3}  = responsePep,
%   peptide{ii,4}  = numZPep,
%   peptide{ii,5}  = idCluster;
%   peptide{ii,6}  = indHit;
%   peptide{ii,7}  = rtTable;       % After pass 1
%   peptide{ii,8}  = rtPosDelta;    % After pass 1
%   peptide{ii,9}  = rtNegDelta};   % After pass 1
%   peptide{ii,10} = retTimePepTarget;  % ret time in target
%
%   param.retTimeOffsetTracking = dinput('Retention time offset for tracking pass (min)',5 ✔
.0);
%   param.heightRatioTracking   = dinput('Height ratio range for tracking pass ',0.4);
%   param.SNRLimitTracking      = dinput('SNR Limit for tracking pass',10);
%   param.mzOffset              = dinput('Maximum m/z offset (amu)',0.02);
%   param.stopForAllPlots       = dinput('Stop for all plots?',0);
%   param.zThreshold = 0.5;

function peptide = track3DLoop01 (peptide,param);

disp(['Execute: ',mfilename])

mwHPlusTarget   = peptide{param.iiTarget,1};
retTimeTarget   = peptide{param.iiTarget,2};
responseTarget  = peptide{param.iiTarget,3};
numZTarget      = peptide{param.iiTarget,4};

for ii = 1:length(responseTarget)

    if rem(ii,500)==0
        disp(ii)
    end

    if responseTarget(ii) < param.SNRLimitTracking
        continue;
    end
    thisRetTime  = retTimeTarget(ii);
    thisMwHPlus  = mwHPlusTarget(ii);
    thisResponse = responseTarget(ii);
    thisNumZ     = numZTarget(ii);

    for jj = 1: param.numInjections
        if jj == param.iiTarget
            continue
        end
        mwHPlusA    = peptide{jj,1};
        retTimeA    = peptide{jj,2};
        responseA   = peptide{jj,3};
        numZA       = peptide{jj,4};

        indSubset   = find(abs(retTimeA-thisRetTime)<param.retTimeOffsetTracking);
```

```
% [peptide] = track3DPassB01 (peptide,param);
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%    Storage map
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%  peptide{ii,1}  = mwHPlusPep;
%  peptide{ii,2}  = retTimePep,
%  peptide{ii,3}  = responsePep,
%  peptide{ii,4}  = numZPep,
%  peptide{ii,5}  = idCluster;
%  peptide{ii,6}  = indHit;
%  peptide{ii,7}  = rtTable;        % After pass 1
%  peptide{ii,8}  = rtPosDelta;     % After pass 1
%  peptide{ii,9}  = rtNegDelta};    % After pass 1
%  peptide{ii,10} = retTimePepTarget;  % ret time in target
%  peptide{ii,11} = indHit;            % defined in PassB
%
%  param.retTimeOffsetTracking = dinput('Retention time offset for tracking pass (min)',5
.0);
%  param.heightRatioTracking   = dinput('Height ratio range for tracking pass ',0.4);
%  param.SNRLimitTracking      = dinput('SNR Limit for tracking pass',10);
%  param.mzOffset              = dinput('Maximum m/z offset (amu)',0.02);
%  param.stopForAllPlots       = dinput('Stop for all plots?',0);
%  param.zThreshold = 0.5;

% Definition
%     param.sigmaRetTimeResid(ii) = median(abs(retTimeResidNoZero))/0.67;


function peptide = track3DPassB01 (peptide,param);

disp(['Execute: ',mfilename])

mwHPlusTarget  = peptide{param.iiTarget,1};
retTimeTarget  = peptide{param.iiTarget,2};
responseTarget = peptide{param.iiTarget,3};
numZTarget     = peptide{param.iiTarget,4};

for ii = 1:length(responseTarget)

    if rem(ii,500)==0
        disp(ii)
    end

    if responseTarget(ii) < param.SNRLimitID
        continue;
    end

    if numZTarget(ii) < param.zLowerLimit | numZTarget(ii) > param.zUpperLimit
        continue;
    end

    thisRetTime  = retTimeTarget(ii);
    thisMwHPlus  = mwHPlusTarget(ii);
    thisResponse = responseTarget(ii);
    thisNumZ     = numZTarget(ii);
```

93

ABSTRACT

Ion Detection in Three Dimensions: A Novel Algorithm to Detect and Quantify Ions Obtained from High-Accuracy LC/MS Separations of Tryptic Digests of Complex Protein Mixtures

Introduction: The potential of LC/MS separations of complex mixtures is fully realized only when all the ions detected by the mass spectrometer are recovered in the analysis of the data. Once detected, the ions can be used for quantitative and qualitative purposes. Ions from isotopes of peptides, for example, can assembled into clusters and their mono-isotope mass can be accurately determined.

Thus the deceptively simple problem of ion detection is in fact, a potential limiting step in the exploitation of LC/MS data. For example, a peak-detection algorithm originally designed to detect peaks in a spectrum may be adopted to address the problem of ion detection in three-dimensional LC/MS separations, resulting in less than optimal performance. Here, we introduce a novel three-dimensional ion detection algorithm optimized for the analysis of high-mass-accuracy LC/MS data.

Methods: The method assembles the spectra obtained from the LC/MS separation into a matrix. The columns of the matrix are the spectra, the rows are the chromatograms. A novel convolution method, based on the properties of matched filters is applied to this matrix. The properties of the filter are designed to identify all the potentially detectable ions present in the data. Thus the approach lends itself to resolution enhancement: Pairs of ions that are only partially resolved or appear as shoulders, can separately quantified. In addition, low-intensity ions that might otherwise be overlooked can be detected; thus the method lends itself to the analysis of samples whose intensities span the full dynamic range of the instrument.

Results: The samples used to evaluate this new algorithm were obtained from tryptic digests of proteins test mixtures and from serum spiked with the test mixtures. We obtained data from these digests using a high-resolution (>17,000) orthogonal quadrupole time of flight mass spectrometers. The ions resulting from different isotopic states are separately detected and high accuracy mass, retention time, and intensity values for each ion are obtained. These cluster-associated ions are assembled into clusters producing unique value for mwHPlus for each cluster.

We demonstrate the quantitative reproducibility of molecular weight, retention time, and intensity of the data over the large dynamic range of this data as obtained using this algorithm.

Statistical study of LC/MS/MS data of human serum

This work provides a statistical study of LC/MS/MS data of human serum. The statistical study is very important for understanding the experimental data of complex biological mixtures. The digested peptides from sample (human serum) are run by LC/MS/MS. The raw data from LC/MS/MS are processed to generated ion sticks. Each ion stick has three parameters: m/z, retention time and intensity. One dimensional histograms of m/z, retention time and intensity for both MS data and MS/MS data are studied. Two dimensional histogram of m/z, retention time for both MS and MS/MS data are also studied. By those studies we can find what are most frequent m/z, retention time and intensity. Then the ion sticks are deconvoluted into peptide lists by both charge and isotopic deconvolution for both MS data and MS/MS data. Each peptide is come from multiple isotopes and multiple charges. Each peptide has three parameters: peptide mhp, peptide retention time and peptide intensity. The histograms of peptide mhp, peptide retention time and peptide intensity are studied. The most frequent peptide mhp, peptide retention time and peptide intensity are found. Two dimensional histogram of peptide's mhp, retention time for both MS and MS/MS data are also studied for different number of bins of mhp and retention time. This indicates how many peptides can be found in a certain mhp and retention time window for the complex biological mixtures.

Next step is to study the replication of injections. Sample (Human serum) is run by LC/MS/MS for replicated three times. The statistical calculations (mean, median, stand deviation and coefficient of variation) for number of peptides and total peptide intensities from 3 injections of the sample are provided. The coefficient of variation of number of peptides and total peptide intensities between 3 injections of the sample is about 5%. This demonstrates the reproducibility of total number of peptides and total peptide intensities. In summary we have studied the statistics of LC/MS/MS data of human serum, which is very useful for understating the experimental data of LC/MS/MS of complex biological mixtures.

Statistical study of LC/MS data of human serum spiked with five proteins

This work provides a statistical study of LC/MS data of human serum spiked with five proteins. The statistical study is an important step for quantitatively compare the relative level of proteins contained in two or more complex biological mixtures. Two samples are used for this study: sample 1 has human serum spiked with 5 pmole five proteins, sample 2 has human serum spiked with 1 pmole five proteins. The digested peptides from samples are run by LC/MS. Each sample has 3 replicated LC/MS runs. The raw data from LC/MS are processed to generated ion sticks. Each ion stick has three parameters: m/z, retention time and intensity. Then the ion sticks are deconvoluted into peptide sticks by both charge and isotopic deconvolution. Each peptide is come from multiple isotopes and multiple charges. Each peptide has three parameters: peptide mhp, peptide retention time and peptide intensity. The statistical calculations (mean, median, stand deviation and coefficient of variation) for number of peptides and total peptide intensities from 3 injections of each sample are provided. This demonstrates the reproducibility of total number of peptides and total peptide intensities.

Next step is to study the replication of each peptide from 3 injections of each sample. Number of replicated peptides and replicated intensities are studied. About 60% of peptides and about 90% of peptides intensities are replicated. This indicates the non-replicated peptides are small intensity one. The average of coefficient of variation of replicated intensities is about 20%. Then the replications of each peptide from 6 injections of two samples are studied. For all the peptides which are replicated for 6 times, the statistical calculations (mean, median, stand deviation and coefficient of variation) of peptide intensities from 3 injections of each sample are provided. The mean intensities of replicated peptides between two samples are compared to indicate the relative level change of spiked proteins contained in two samples. The ratio of mean intensities of replicated peptides between two samples is plot against mean coefficient of variation of intensities of two samples. In summary we have done statistical study of LC/MS data, which is very useful for quantitatively compare of the relative level of proteins contained in two or more complex biological mixtures.

Statistical study of LC/MS/MS data of human serum

This work provides a statistical study of LC/MS/MS data of human serum. The statistical study is very important for understanding the experimental data of complex biological mixtures. We study two cases: case 1 for one injection of one sample, case 2 for three or more injections of one or two samples.

Case 1 studies one injection of one sample: This study provides the statistics of ions and peptides of LC/MS/MS data of human serum. The digested peptides from human serum are run by LC/MS/MS for one time. The raw data from LC/MS/MS are processed to generated ion sticks. Each ion stick has three parameters: m/z, retention time and intensity. One dimensional histograms of m/z, retention time and intensity for both MS data and MS/MS data are studied. Two dimensional histogram of m/z, retention time for both MS and MS/MS data are also studied. By those studies we can find what are most frequent m/z, retention time and intensity. Then the ion sticks are deconvoluted into peptide lists by both charge and isotopic deconvolution for both MS data and MS/MS data. Each peptide is come from multiple isotopes and multiple charges. Each peptide has three parameters: peptide mhp (peptide mass plus proton mass), peptide retention time and peptide intensity. The histograms of peptide mhp, peptide retention time and peptide intensity are studied. The most frequent peptide mhp, peptide retention time and peptide intensity are found. Two dimensional histogram of peptide's mhp, retention time for both MS and MS/MS data are also studied for different number of bins of mhp and retention time. This indicates how many peptides can be found in a certain mhp and retention time window for the complex biological mixtures.

Case 2 studies three or more injections of one or two samples: This study provides the statistics of LC/MS replicated data of one or two samples. The statistical calculations (mean, median, stand deviation and coefficient of variation) for number of peptides and total peptide intensities from 3 injections of the sample (human serum) are provided. About 60% of peptides and about 90% of peptides intensities are replicated. For all the replicated peptides, histograms of mhp difference, retention time difference and intensity difference of replicated peptide's pair are studied. The statistical calculations (mean, stand deviation and coefficient of variation) for mhp and intensity of replicated peptides are also provided. Histograms of mean intensity of replicated peptides and non-replicated peptides are also studied. Similar statistical study for six injections of two samples (sample 1 has human serum spiked with 5 pmole five proteins, sample 2 has human serum spiked with 1 pmole five proteins) is also provided. The mean intensities of replicated peptides between two samples are compared to indicate the relative level change of spiked proteins contained in two samples.

Towards Quantitative Global Proteomics: Statistical Results Obtained from Multiple Tryptic Digests of Complex Protein Mixtures Using Novel Algorithms for the Detection, Tracking and Quantitation of Peptides

Introduction: Quantitation of proteins by high-mass accuracy LC/MS separations requires reproducible sample preparation, robust separation methods, and accurate mass measurements. With such high quality such data in hand, our attention must turn to the algorithms needed to extract information from this data. One critical algorithmic step is reliable tracking of molecular entities between samples.

A molecular entity detected in one injection could be located (i.e, tracked) in another injection by comparing only mass values. However, in the case of complex mixtures, such as tryptic digests, a retention-time search-window of a few minutes may contain pairs of entities that have the same *measured* mass, but in fact are unrelated. The resulting mistakes in tracking will compromise quantitation.

Methods: The novel algorithmic method introduced here addresses the problem of tracking. The method relies on accurate mass measurements to find the subset of entities that can be uniquely tracked by accurate mass alone. These unique matched pairs determine a retention time map, and such a map is found for all injections in a sample set.

These maps are then used to assign a unique reference retention time to *all* molecular entities in *all* injections. The method used the unique paired masses as, in effect, internal standards to correct for the retention time offset of all entities. The reference retention times of an entity can then be compared between any two samples in the sample set.

Results: The reference retention time puts all samples on an equal footing. The search window associated with the reference retention time can be as low as +/-0.2 minutes, much smaller than conventional minutes wide search windows. The reference retention time together with accurate mass can then be used to track an entity from injection to injection in a sample set.

Tryptic digests that contain upwards of 10,000 unique masses whose nearly 100,000 ions can be detected in a 2-hour LC separation followed by online MS detection.

4) TITLE:

Protocols to Assure Reproducible Quantitative and Qualitative Analysis of Tryptic Digests of Complex Protein Mixtures for Global Proteomic Experiments

Introduction: Meaningful results in qualitative and quantitative proteomics, such as observation of differing expression levels of a protein in a series of samples, can only be obtained if samples are consistently prepared and analyzed. Tryptic digestion must be carried to completion for all proteins in order to maximize sequence coverage for identification and to all meaningful quantitative sample-to-sample comparison of a given peptide. Chromatographic separation of the resulting mixtures must also be performed in a consistent manner.

We have developed protocols for tryptic digestion of protein mixtures designed to assure reproducible peptide production, protocols to assure maximum reproducibility of capillary scale HPLC, and software tools to easily verify the reproducibility of our experiments.

Methods: A series of replicate digests of commercial rat serum was prepared. A proprietary detergent (RapiGest™ SF, Waters Corporation) was used as a denaturating agent. One or more standardized tryptic digests of individual proteins (MassPrep™ Digestion Standards, Waters Corporation) were added to the digests. Samples were analyzed by direction onto a 300 micron diameter x 15 cm column packed with Atlantis™ dC_{18} packing and eluted with a water/acetonitrile/formic acid gradient. The column effluent was directed a Nano Lockspray source on a hybrid quadrupole-time of flight mass spectrometer (Q-ToF Ultima API, Waters Corporation) Mass spectral data was obtained alternating scans of low and high collision cell energy. Every 10 seconds a separate reference sample spectrum was obtained.

Results: Use of the detergent as a denaturating agent was found not to interfere with chromatography or ionization of the tryptic peptides, nor was there any observable fouling of the ion source.

Sample consistency was demonstrated as follows: Raw mass spectral data was processed by Protein Lynx Global Server (Waters Corporation) to compile a list of data points as pairs of retention times and accurate mass values (observed m/z values at that moment corrected by use of the reference mass channel, accurate to 10 ppm or less). The resulting data are compared by submission to a software tool (Track 3D, Waters Corporation, patent pending) which correlates retention time, accurate mass values, and signal intensities of two or more samples. Results of this correlation show that signals for a given mass are observed at similar retention time from sample to sample for a great plurality of the observed signals as demonstrated on a graphical representation of difference in retention time vs. retention time for any pair of data sets. Furthermore, we observe that data that replicates in such a fashion represents a very high percentage of the total ion signal intensity for all the data in question, thus demonstrating reproducibility from sample to sample.

Fuller details of our protocols will be included in the poster.

FIG 1

2415 Peptides in Inj 2 hit 3258 Peptides in target with SNR > 100. RetTime Outliers = 115

ΔT_{Rel} (min)

Ret time (min) of target

FIG 2

2415 Peptides in Inj 2 hit 3258 Peptides in target with SNR > 100. RetTime Outliers = 115

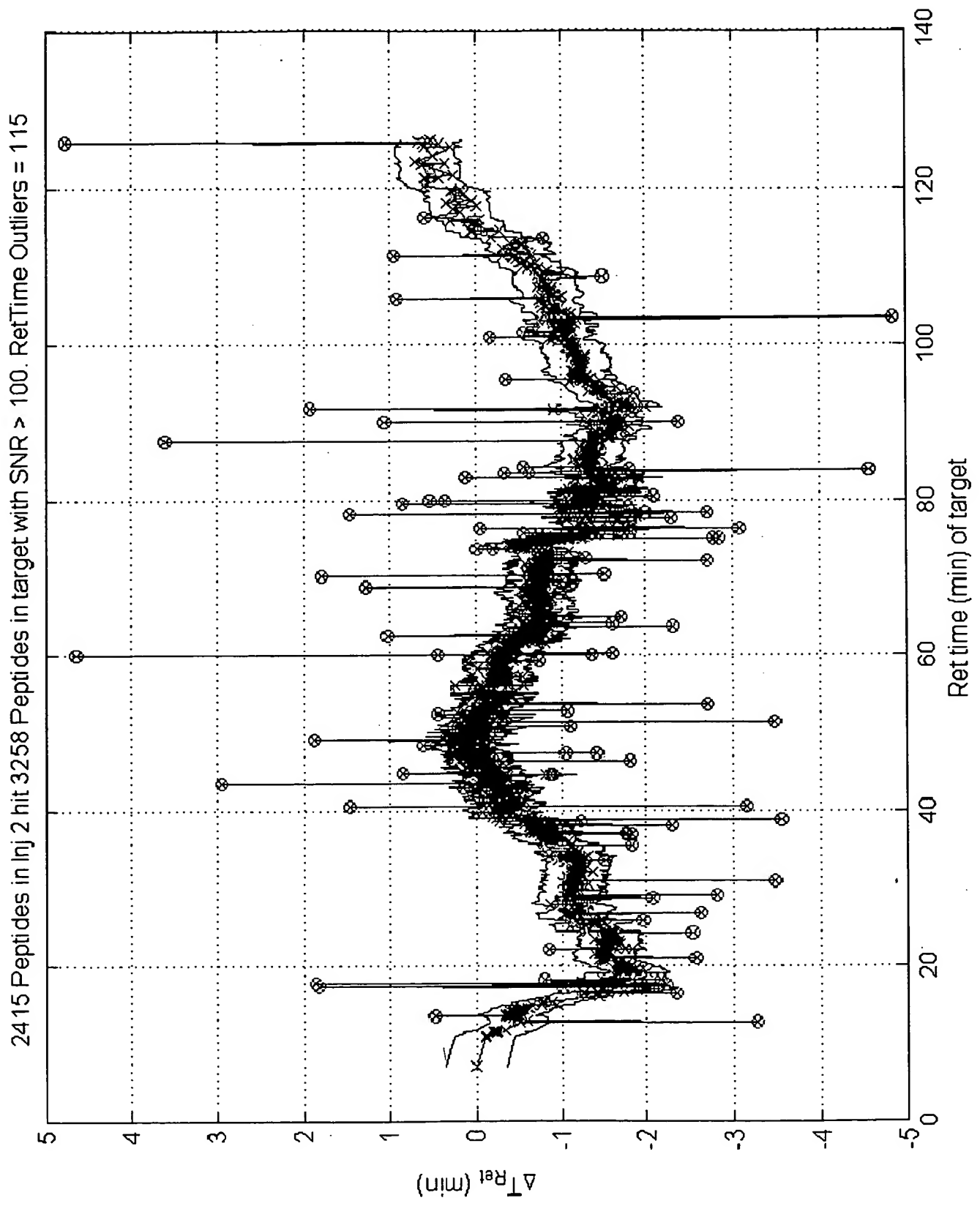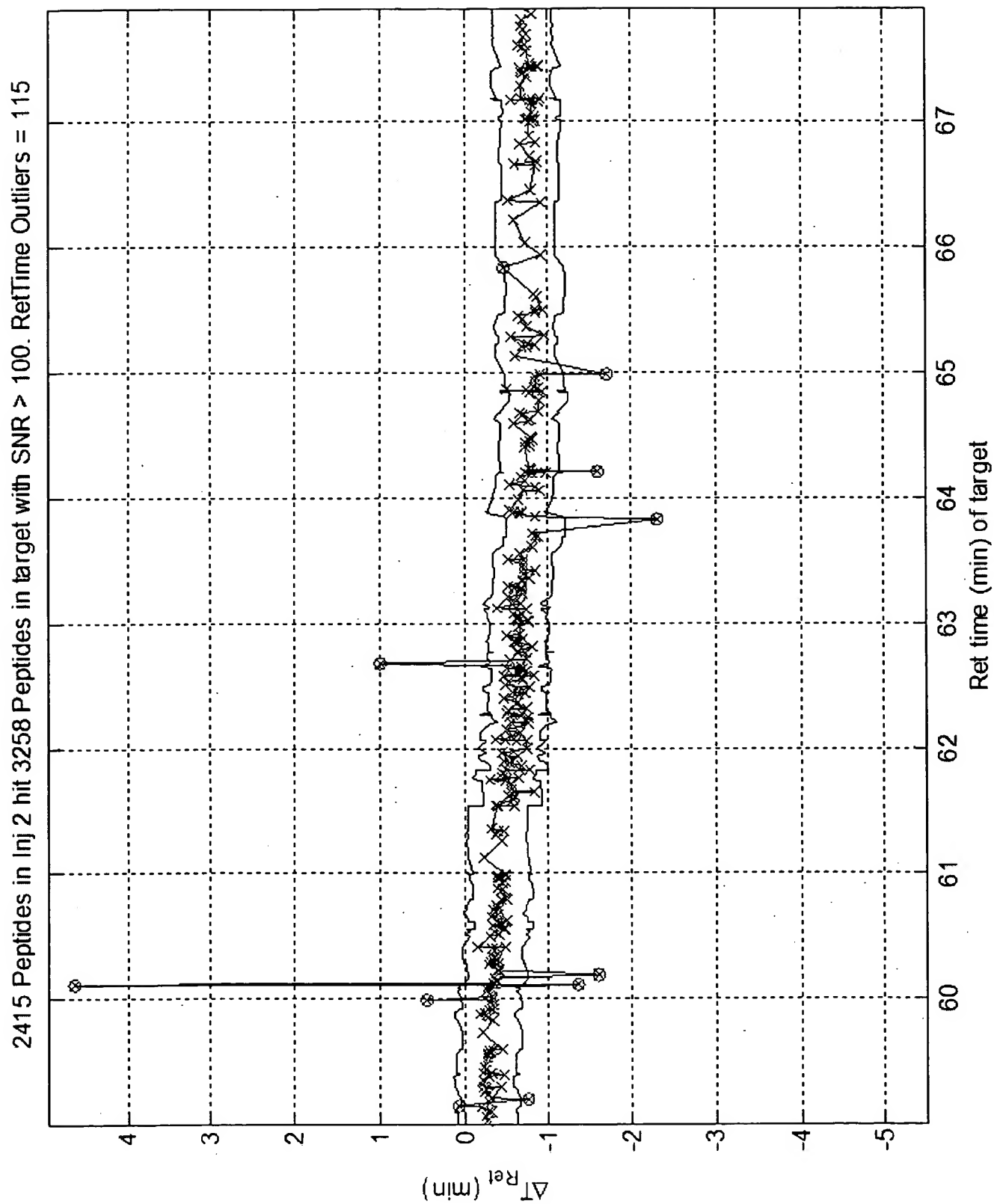ΔT_{Ret} (min)

Ret time (min) of target

FIG 3

Pass2: 7202 Peptides from Injection 2 hit 22027 Peptides in target with SNR > 10. RetTime Limits = ± 0.36 min
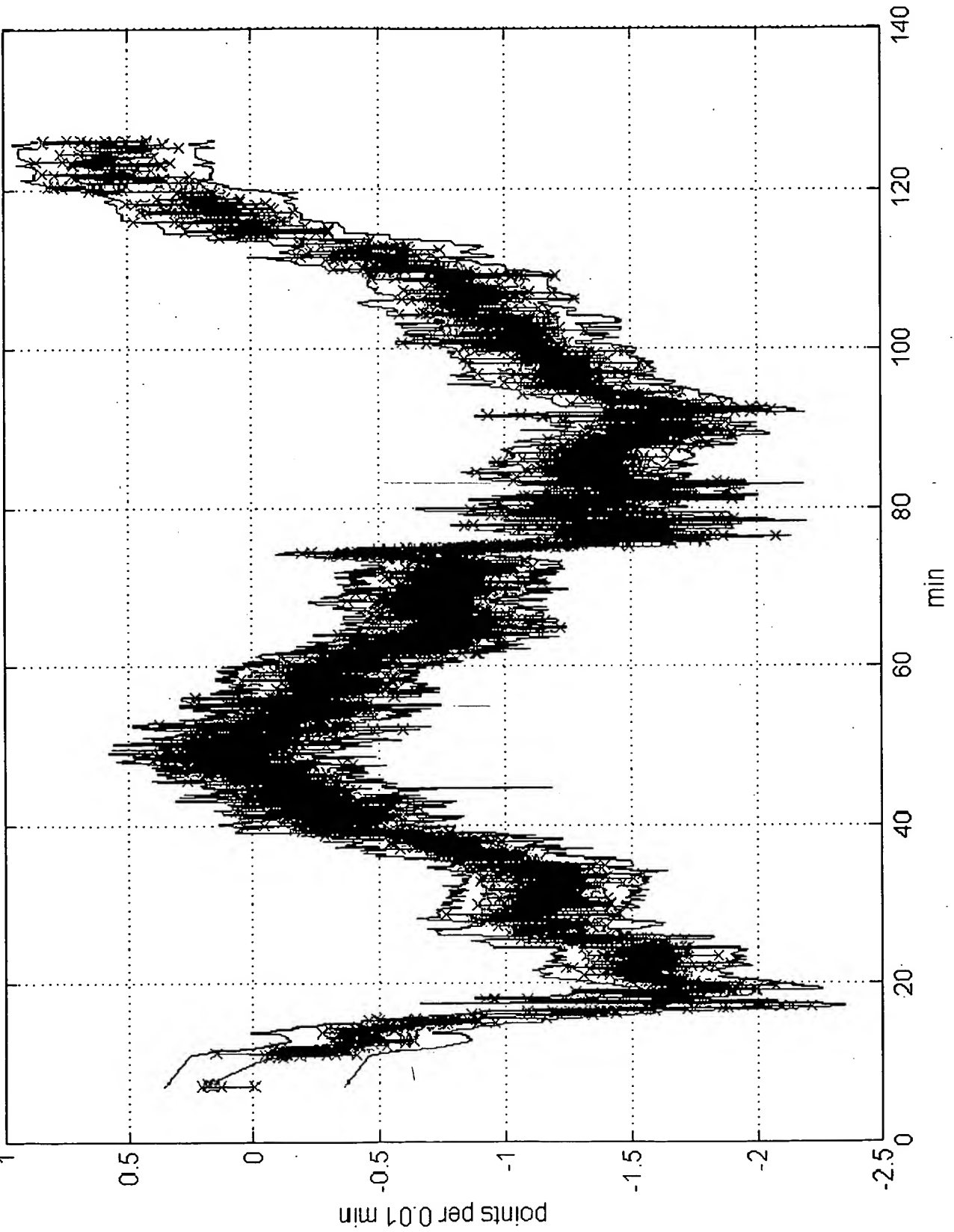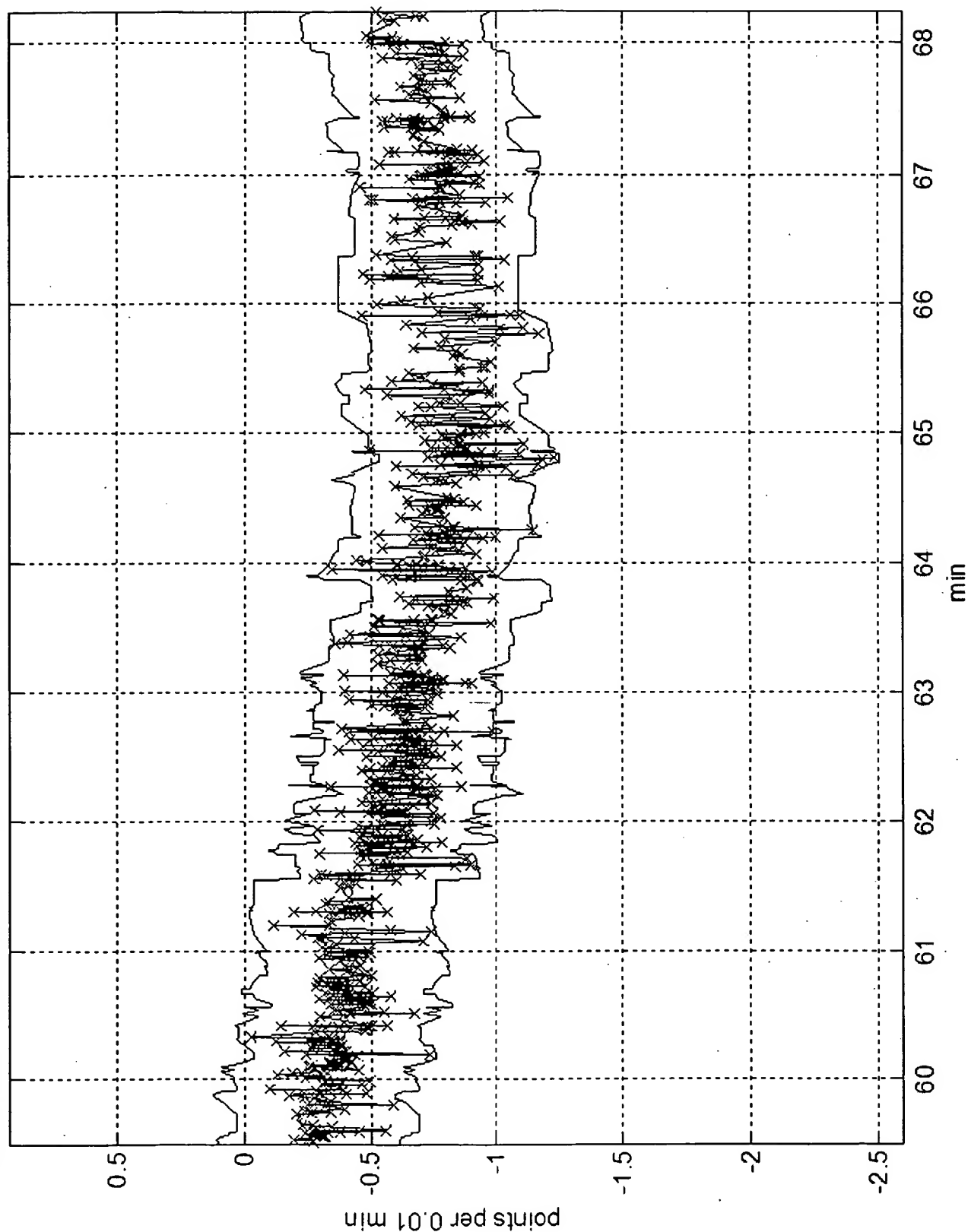
points per 0.01 min

min

FIG 4

Pass2 : 7202 Peptides from Injection 2 hit 22027 Peptides in target with SNR > 10. RetTime Limits = ± 0.36 min

FIG 5 04-388

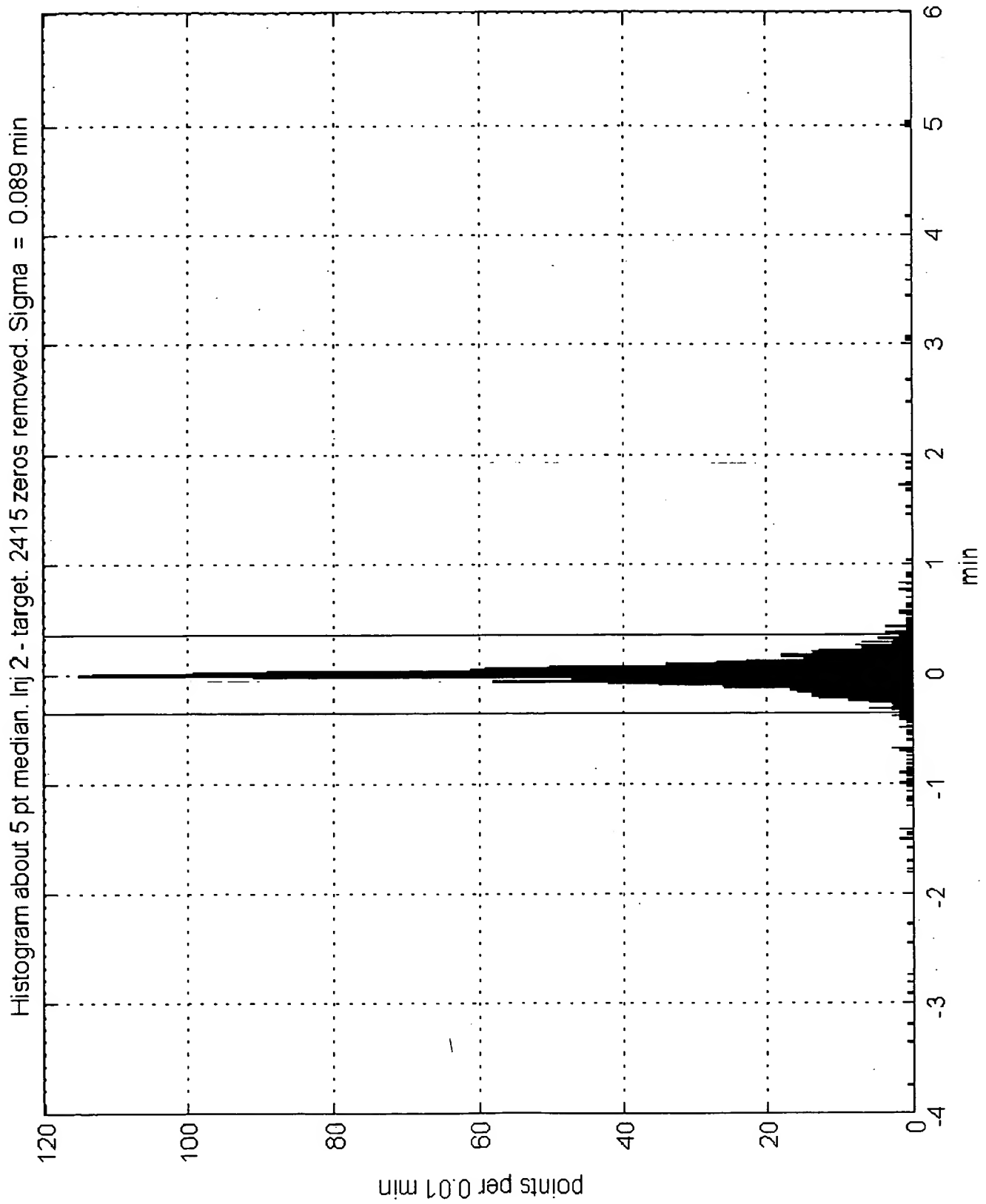Histogram about 5 pt median. Inj 2 - target. 2415 zeros removed. Sigma = 0.089 min

points per 0.01 min

min

FIG 6

2415 peptides. SNR> 100. Inj 2. Ret time ± 0.089 min. Sigma = 0.0052 amu.

Inj 2. Sigma = 3.0 ppm.